Click **here** to ask an associate for production status of specific part numbers.

DS28E18

1-Wire® to I²C/SPI Bridge with Command Sequencer

General Description

The DS28E18 is a simple communications bridge that resides at a remote SPI or I²C sensor and allows the sensor to be controlled by just two wires coming from the host system. It reduces the wire count from six (for SPI) or four (for I²C). These two wires use Analog Devices' 1-Wire protocol that combines power and signal on a single wire, and which is driven by the programmable I/O pins on the host's microcontroller. The 1-Wire network supports connection lengths up to 100m and 10 sensor nodes or more.

The IC provides a 512-byte command sequencer in SRAM that can be loaded with multiple I²C or SPI commands. Once loaded, the host controller sends a command to execute the sequence, power, and collect data from attached I²C or SPI peripherals. A subsequent 1-Wire command reads collected data. Power for attached sensors or peripherals is sourced from the 1-Wire line making the DS28E18 a very efficient solution to remotely power and control complex I²C or SPI devices such as sensors, ADCs, DACs, and display controllers.

When used as a bridge for I²C slave devices, the DS28E18 communicates at Standard mode (100kHz), Fast mode (400kHz) or Fast-mode Plus (Fm+, 1MHz). In SPI mode, multiple clock rates are supported up to 2.3MHz. Configuring for I²C or SPI operation is performed with a 1-Wire command; I²C is the power-on default. When operating in I²C mode, two programmable GPIO pins are available for additional peripheral control.

Each DS28E18 provides a unique and secure 64-bit ROM identification number (ROM ID) that serves as the device's address on the 1-Wire bus. Multiple DS28E18 devices can coexist with other devices in a 1-Wire network and be accessed individually without affecting other devices.

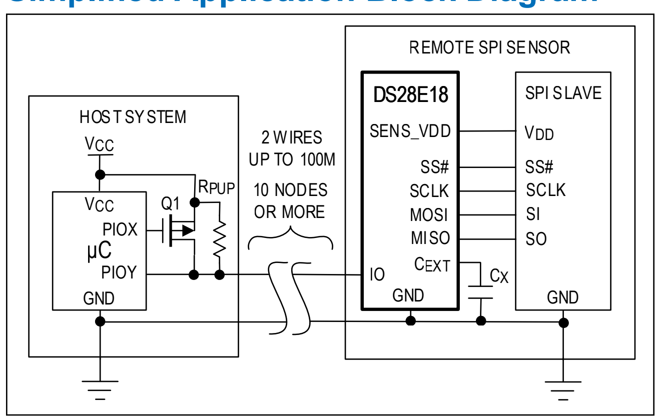
Applications

- Examining Environmental Conditions
- Accessory Identification and Control
- Equipment Configuration and Monitoring
- Grain Elevator Monitoring

Benefits and Features

- Operate Remote I²C or SPI Devices Using Single-Contact 1-Wire Interface
  - Extending I²C/SPI Communication Distance
  - Reduce Six Wires (for SPI) or Four Wires (for I²C) to Two Wires
  - 512-Byte Sequencer for Autonomous Operation of Attached Devices
  - Two Configurable GPIO Pins for Additional Peripheral Control
- No External Power Required
  - DS28E18 Parasitically Powered from 1-Wire
  - I²C/SPI Peripheral Power Derived from the 1-Wire Line
- Flexible 1-Wire and I²C/SPI Master Operational Modes
  - Supports Standard (11kbps) and Overdrive (90kbps) 1-Wire Communication
  - 100kHz, 400kHz, and 1MHz for I²C Slaves
  - Up to 2.3MHz for SPI Slaves
- Easy to Integrate
  - Small, 2mm x 3mm x 0.75mm, 8-Pin TDFN Package
  - -40°C to +85°C Operation
  - 2.97V to 3.63V Operating Voltage Range

Simplified Application Block Diagram



1-Wire is a registered trademark of Maxim Integrated Products, Inc.

Ordering Information appears at end of data sheet.

19-100832; Rev 1; 1/24

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to GND .... -0.5V to 4.0V
Maximum Current into Any Pin........................... -20mA to 20mA
Continuous Power Dissipation (Single-Layer Board) (T_A = +70°C, derate 16.70mW/°C above +70°C)........................... 1333.30mW
Continuous Power Dissipation (Multilayer Board) (T_A = +70°C, derate 16.70mW/°C above +70°C.).......................... 1333.30mW
Operating Temperature Range ....-40°C to +85°C
Junction Temperature .... +125°C
Storage Temperature Range ....-40°C to +125°C
Soldering Temperature (reflow) .... +260°C
Lead Temperature (soldering, 10s)................................. +260°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Package Information

8 TDFN-EP

Package Code	T823+3C
Outline Number	21-0174
Land Pattern Number	90-0091
Thermal Resistance, Single-Layer Board:	
Junction to Ambient (θ_{JA})	60°C/W
Junction to Case (θ_{JC})	11°C/W
Thermal Resistance, Four-Layer Board:	
Junction to Ambient (θ_{JA})	60°C/W
Junction to Case (θ_{JC})	11°C/W

For the latest package outline information and land patterns (footprints), go to the *Package Index* on the Analog Devices website. Note that a "+", "#", or "-" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to *Thermal Characterization of IC Packages*.

Electrical Characteristics

(Limits are 100% tested at T_A = +25°C and T_A = +85°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and characterization. Specifications marked GBD are guaranteed by design and not production tested. Specifications to the minimum operating temperature are guaranteed by design and are not production tested.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
IO PIN: GENERAL DATA						
1-Wire Pullup Voltage	V_{PUP}	(*Note 1*)	2.97		3.63	V
1-Wire Pullup Resistance	R_{PUP}	(*Note 1*, *Note 2*)	300		1000	Ω
Input Capacitance	C_{IO}	(*Note 4*)		0.1 + C_{EXT}		nF
Capacitor External	C_{EXT}	(*Note 1*)	399.5	470		nF
Voltage Capacitor External Min	V_{CEXT}	t_{RSTL} = 640µs		1.5		V
Input Load Current	I_L	IO pin at V_{PUP}		8.5	300	µA
High-to-Low Switching Threshold	V_{TL}	(*Note 3*, *Note 5*, *Note 6*)		0.65 x V_{PUP}		V

Electrical Characteristics (continued)

(Limits are 100% tested at T_A = +25°C and T_A = +85°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and characterization. Specifications marked GBD are guaranteed by design and not production tested. Specifications to the minimum operating temperature are guaranteed by design and are not production tested.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Input Low Voltage	V_{IL}	(Note 7)				0.1 x V_{PUP}	V
Low-to-High Switching Threshold	V_{TH}	(Note 3, Note 5, Note 8)			0.75 x V_{PUP}		V
Switching Hysteresis	V_{HY}	(Note 3, Note 5,Note 9)			0.3		V
Output Low Voltage	V_{OL}	I_{OL} = 4mA (Note 10)				0.4	V
Recovery Time	t_{REC}	R_{PUP} = 1000Ω, (Note 1, Note 11, Note 12)	Standard speed	25			µs
			Standard speed, directly prior to reset pulse	100			
			Overdrive speed	5			
			Overdrive speed, directly prior to reset pulse	10			
Rising-Edge Hold-off Time	t_{REH}	Applies to standard speed only (Note 3, Note 13)			1		µs
Time Slot Duration	t_{SLOT}	Standard speed (Note 3, Note 14)		85			µs
		Overdrive speed (Note 3, Note 14)		11			
IO PIN: 1-Wire RESET, PRESENCE-DETECT CYCLE							
Reset Low Time	t_{RSTL}	Standard speed (Note 1)		480		640	µs
		Overdrive speed (Note 1)		48		80	
Reset High Time	t_{RSTH}	Standard speed (Note 1, Note 15)		480			µs
		Overdrive speed (Note 1, Note 15)		48			
Presence-Detect Fall Time	t_{FPD}	Standard speed (Note 3, Note 16)			1.25		µs
		Overdrive speed (Note 3, Note 16)			0.15		
Presence-Detect Sample Time	t_{MSP}	Standard speed (Note 1, Note 17)		65		75	µs
		Overdrive speed (Note 1, Note 17)		7		10	
IO PIN: 1-Wire WRITE							
Write-Zero Low Time	t_{W0L}	Standard speed (Note 1, Note 18)		60		120	µs
		Overdrive speed (Note 1, Note 18)		6		16	
Write-One Low Time	t_{W1L}	Standard speed (Note 1, Note 18)		0.25		15	µs
		Overdrive speed (Note 1, Note 18)		0.25		2	
IO PIN: 1-Wire READ							
Read Low Time	t_{RL}	Standard speed (Note 1, Note 19)		0.25		15 - δ	µs
		Overdrive speed (Note 1, Note 19)		0.25		2 - δ	
Read Sample Time	t_{MSR}	Standard speed (Note 1, Note 19)		t_{RL} + δ		15	µs
		Overdrive speed (Note 1, Note 19)		t_{RL} + δ		2	
STRONG PULLUP OPERATION							
Strong Pullup Current	I_{SPU}	SENS_VDD Off				4	mA
		I_{SENS_VDD} = 10mA (Note 20)				14	

Electrical Characteristics (continued)

(Limits are 100% tested at T_A = +25°C and T_A = +85°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and characterization. Specifications marked GBD are guaranteed by design and not production tested. Specifications to the minimum operating temperature are guaranteed by design and are not production tested.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Strong Pullup Voltage	V_{SPU}	(Note 20)	2.0			V
SENSOR V_{DD} OUTPUT SUPPLY PIN						
SENS_VDD Output Voltage	V_{SENS_VDD}	Sequencer active, IO = 3.3V, I_{SENS_VDD} = 1mA, R_{LOAD} = 3.3kΩ, SENS_VDD on (Note 20)		3.28		V
SENS_VDD Current	I_{SENS_VDD}	Sequencer active, IO = 3.3V, SENS_VDD on (Note 20)			10	mA
SENS_VDD High-Z		Strong pullup not active or SENS_VDD off		10		MΩ
I2C, SPI, AND GPIO PINS						
Output Low	PIO_V_{OL}	I_{OL} = 4mA (Note 10)			0.4	V
Output High	PIO_V_{OH}	I_{OH} = -2mA	V_{SPU} - 0.4			V
Input Low	PIO_V_{IL}		-0.3		0.18 x V_{SPU}	V
Input High	PIO_V_{IH}		0.70 x V_{SPU}		V_{SPU} + 0.3	V
Switching Hysteresis	PIO_V_{HY}			0.05 x V_{SPU}		V
Leakage Current	PIO_I_L	Not including any pullup/pulldown current.	-1		+1	µA
Input Capacitance	PIO_C_I	(Note 3)		10		pF
Command Timing						
Operation Time	t_{OP}	Note 3			1	ms
I²C MASTER (STANDARD MODE) (Note 22)						
SCL Clock Frequency	f_{SCL}				100	kHz
SCK High Time	t_{HIGH}		4.0			µs
SCK Low Time	t_{LOW}		4.7			µs
Start Setup Time	$t_{SU:STA}$		4.7			µs
Stop Setup Time	$t_{SU:STO}$		4.0			µs
Data Setup Time	$t_{SU:DAT}$		250			ns
Data Hold Time	$t_{HD:DAT}$	(Note 23)	0			µs
Start Hold Time	$t_{HD:STA}$		4			
Bus Free Time Between STOP and START	t_{BUF}		4.7			µs
Fall Time	t_F			30		ns
Capacitive Load for Each Bus Line	C_B	I²C mode (Note 1, Note 24)			400	pF
I²C MASTER (FAST MODE) (Note 22)						
SCL Clock Frequency	f_{SCL}				400	kHz
SCK High Time	t_{HIGH}		0.6			µs
SCK Low Time	t_{LOW}		1.3			µs

Electrical Characteristics (continued)

(Limits are 100% tested at T_A = +25°C and T_A = +85°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and characterization. Specifications marked GBD are guaranteed by design and not production tested. Specifications to the minimum operating temperature are guaranteed by design and are not production tested.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Start Setup Time	$t_{SU:STA}$		0.6			µs
Stop Setup Time	$t_{SU:STO}$		0.6			µs
Data Setup Time	$t_{SU:DAT}$		100			ns
Data Hold Time	$t_{HD:DAT}$	(Note 23)	0			µs
Start Hold Time	$t_{HD:STA}$		0.6			µs
Bus Free Time Between STOP and START	t_{BUF}		1.3			µs
Fall Time	t_F			30		ns
Capacitive Load for Each Bus Line	C_B	I^2C mode (Note 1, Note 24)			400	pF
I^2C MASTER (FAST-MODE PLUS) (Note 22)						
SCL Clock Frequency	f_{SCL}				1	MHz
SCK High Time	t_{HIGH}		0.26			µs
SCK Low Time	t_{LOW}		0.5			µs
Start Setup Time	$t_{SU:STA}$		0.26			µs
Stop Setup Time	$t_{SU:STO}$		0.26			µs
Data Setup Time	$t_{SU:DAT}$		50			ns
Data Hold Time	$t_{HD:DAT}$	(Note 23)	0			µs
Start Hold Time	$t_{HD:STA}$		0.26			µs
Bus Free Time Between STOP and START	t_{BUF}		0.5			µs
Fall Time	t_F			30		ns
Capacitive Load for Each Bus Line	C_B	(Note 1, Note 24)			550	pF
SPI MASTER (Note 25)						
SPI Master Operating Frequency	f_{MCK}	SPD = 0b00		88.9	100	kHz
		SPD = 0b01		333.3	400	kHz
		SPD = 0b10		0.8	1	MHz
		SPD = 0b11		2.0	2.3	MHz
SPI Master SCK Period	t_{MCK}			$\frac{1}{f_{MCK}}$		µs
SCK Output Pulse-Width High/Low	t_{MCH}, t_{MCL}			$\frac{t_{MCK}}{2}$		µs
MOSI Output Hold Time After SCK Sample Edge	t_{MOH}			$\frac{3}{4} \times t_{MCK}$		µs
MOSI Output Valid to Sample Edge	t_{MOV}			$\frac{t_{MCK}}{4}$		µs
MISO Input Valid to SCK Sample Edge Setup	t_{MIS}			50		ns

Electrical Characteristics (continued)

(Limits are 100% tested at T_A = +25°C and T_A = +85°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and characterization. Specifications marked GBD are guaranteed by design and not production tested. Specifications to the minimum operating temperature are guaranteed by design and are not production tested.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
MISO Input to SCK Sample Edge Hold	t_{MIH}			50		ns
MOSI Transition to SS Transition	$t_{SS:SU}$			t_{MCK}		µs
SS Active to First SCK Edge	$t_{SS:CLK}$	SPI mode 3		$\frac{3}{4} \times t_{MCK}$		µs

Note 1: System requirement.

Note 2: System requirement. Maximum allowable pullup resistance is a function of the number of 1-Wire devices in the system and 1-Wire recovery times. The specified value here applies to systems with only one device and with the minimum 1-Wire recovery times.

Note 3: Guaranteed by design and/or characterization only. Not production tested.

Note 4: Value represents the internal parasite capacitance when V_{PUP} is first applied. Once the parasite capacitance is charged, it does not affect normal communication. Typically, during normal communication, the internal parasite capacitance is effectively ~100pF.

Note 5: V_{TL}, V_{TH}, and V_{HY} are a function of the internal supply voltage, which is a function of V_{PUP}, R_{PUP}, 1-Wire timing, and capacitive loading on IO. Lower V_{PUP}, higher R_{PUP}, shorter t_{REC}, and heavier capacitive loading all lead to lower values of V_{TL}, V_{TH}, and V_{HY}.

Note 6: Voltage below which, during a falling edge on IO, a logic 0 is detected.

Note 7: The voltage on IO must be less than or equal to V_{ILMAX} at all times the master is driving IO to a logic 0 level.

Note 8: Voltage above which, during a rising edge on IO, a logic 1 is detected.

Note 9: After V_{TH} is crossed during a rising edge on IO, the voltage on IO must drop by at least V_{HY} to be detected as logic 0.

Note 10: The current-voltage (I-V) characteristic is linear for voltages less than 1V.

Note 11: Applies to a single device attached to a 1-Wire line.

Note 12: t_{REC} (min) covers operation at worst-case temperature. V_{PUP}, R_{PUP}, C_{IO}, t_{RSTL}, t_{WOL}, t_{RL}, and t_{RECMIN} can be significantly reduced under less extreme conditions. Contact the factory for more information.

Note 13: The earliest recognition of a negative edge is possible at t_{REH} after V_{TH} has been previously reached.

Note 14: Defines the maximum possible bit rate. Equal to $1/(t_{W0LMIN} + t_{RECMIN})$.

Note 15: An additional reset or communication sequence cannot begin until the reset high time has expired.

Note 16: Time from V_{IO} = 80% of V_{PUP} and V_{IO} = 20% of V_{PUP} at the negative edge on IO at the beginning of the presence detect pulse.

Note 17: Interval after t_{RSTL} during which a bus master can read a logic 0 on IO if there is a device present. The power-up presence detect pulse could be outside this interval, but will be complete within 2ms after power-up.

Note 18: ε in Figure 6 represents the time required for the pullup circuitry to pull the voltage on IO up from V_{IL} to V_{TH}. The actual maximum duration for the master to pull the line low is $t_{W1LMAX} + t_F - \varepsilon$ and $t_{W0LMAX} + t_F - \varepsilon$, respectively.

Note 19: δ in Figure 6 represents the time required for the pullup circuitry to pull the voltage on IO up from V_{IL} to the input-high threshold of the bus master. The actual maximum duration for the master to pull the line low is $t_{RLMAX} + t_F$.

Note 20: I_{SPU} is the current drawn from IO during a strong pullup (SPU) operation. The pullup circuit on IO during the SPU operation should be such that the voltage at IO is greater than or equal to V_{SPUMIN}. A low-impedance bypass of R_{PUP} activated during the SPU operation is the recommended method to meet this requirement. See the *Typical Application Circuits* for details.

Note 21: All I²C timing values are referred to $V_{IH(MIN)}$ and $V_{IL(MAX)}$ levels.

Note 22: See Figure 10 for I²C timing symbol details. Rise and fall times are system dependent and not included.

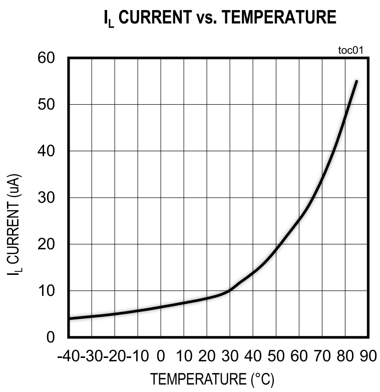
Note 23: The DS28E18 provides 2.5µs (standard mode), 675ns (fast mode), or 280ns (Fm+) minimum hold time, not including rise/fall time, for the SDA signal.

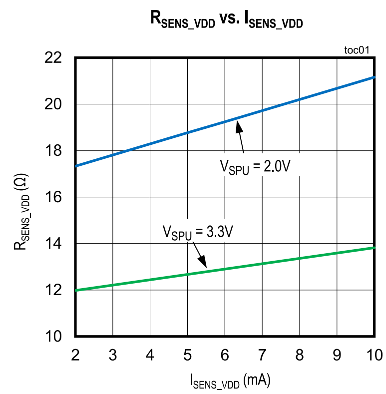
Note 24: C_B = Total capacitance of one bus line in pF. The maximum bus capacitance allowable may vary from this value depending on the actual operating voltage and frequency of the application (I²C bus specification Rev. 03, 19 June 2007).

Note 25: See Figure 12 for SPI timing symbol details. The f_{MCK} options listed only effect speed for the SPI WRITE/READ BYTE command. The f_{MCK} for the SPI WRITE/READ BIT command is variable up to a maximum of 134kHz. Rise and fall times are system dependent and not included.

Typical Operating Characteristics

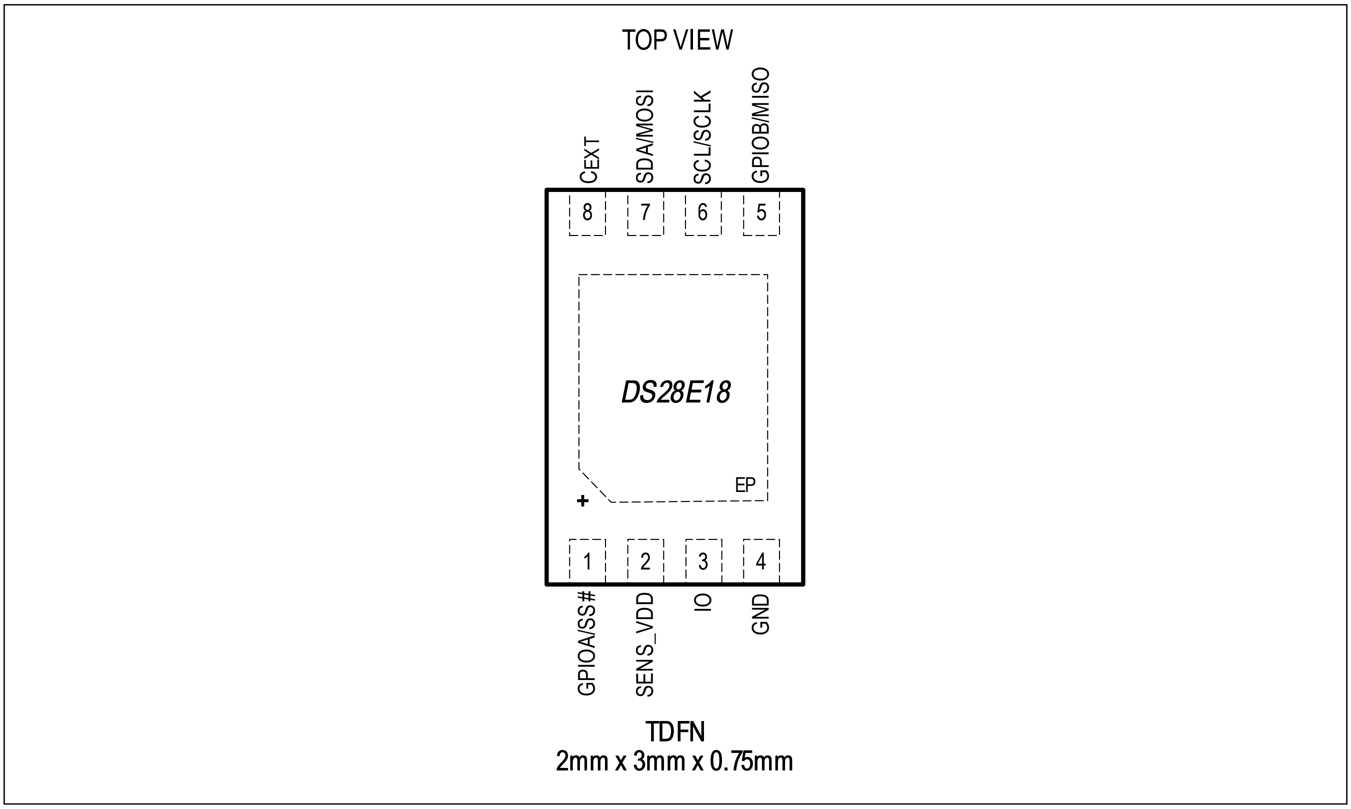
(V_{PUP} = +3.3V; T_A = T_{MIN} to T_{MAX} unless otherwise noted.)





Pin Configuration

DS28E18

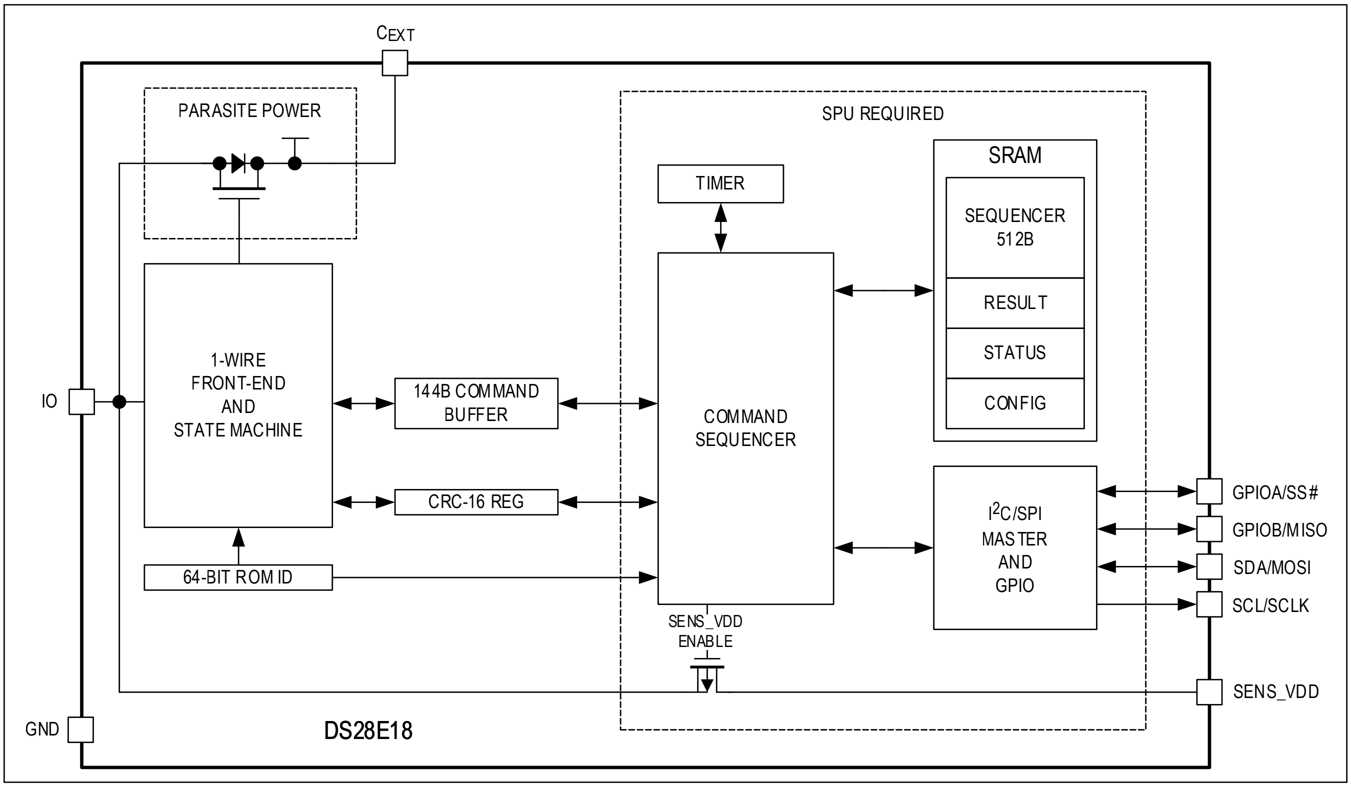


Pin Description

PIN	NAME	FUNCTION
1	GPIOA/SS#	General-Purpose I/O (Default). If the DS28E18 is configured for SPI master operation, this pins is used as an active-low slave select (SS#). In SPI master mode, this pin is output only.
2	SENS_VDD	Output Supply for Powering External I^2C/SPI Sensors/Devices. Connect this output supply pin to the external I^2C/SPI devices power supply pin. This pin is only an output during strong pullup operation. When not in strong pullup operation, this pin is set to a high-impedance state. **Note:** See the *Electrical Characteristics* table for detailed information on the maximum supply current supported.
3	IO	1-Wire Bus Interface. This is an open-drain pin that requires an external pullup resistance (R_{PUP}).
4	GND	Digital Ground
5	GPIOB/MISO	General-Purpose I/O (Default). If the DS28E18 is configured for SPI Master mode, this pin is used as master input slave output (MISO) and operates as an input-only I/O.
6	SCL/SCLK	I^2C Serial Clock (SCL) (Default). If the DS28E18 is configured for SPI Master mode, this pin is used as the SPI clock (SCLK).
7	SDA/MOSI	I^2C Serial-Data Input/Output (Default). If the DS28E18 is configured for SPI Master mode, this pin is used as the master output slave input (MOSI). When configured as MOSI, this pin is an output only.
8	C_{EXT}	Input for External Capacitor. Nominally a 470nF capacitor is to be connected from this pin to ground. This pin acts as the parasite power (i.e., derives power from the 1-Wire bus) during 1-Wire operation. Alternately, this pin may also be directly connected to a power supply in the voltage range of V_{PUP}.
—	EP	Exposed Pad. Solder evenly to the board's ground plane for proper operation. Refer to the *Exposed Pads: A Brief Introduction* application note for additional information.

Functional Diagram

Block Diagram



Detailed Description

The DS28E18 integrates a 1-Wire slave front-end, an I²C/SPI bus master peripheral, GPIO, power control, and functionality to bridge these circuit elements for data communication and power delivery to attached I²C/SPI slaves. The IC has a 144-byte command buffer that utilizes 16-bytes for device function command operations and 128-bytes to transfer formed packets with sequential commands into a 512-byte SRAM sequencer. The formed packets installed in the SRAM sequencer can be called to write and/or read I²C/SPI data to attached slaves. The maximum length of a sequence is 512 bytes. Upon completion of a sequence, the I²C/SPI slave response is recovered using a Read sequencer command. From a host controller, DS28E18 communication is performed serially using the 1-Wire protocol, which requires only a single data connection and a ground return for signaling. The DS28E18 includes a 64-bit unique ROM ID, which guarantees unique and secure identification and also serves as the address of the device in a multidrop 1-Wire network environment where multiple devices reside on a common 1-Wire bus and operate independently of each other.

Figure 1 shows the hierarchical structure of the 1-Wire ROM function, device function, and sequencer commands. The bus master must first provide one of the seven ROM function commands: Read ROM, Match ROM, Search ROM, Skip ROM, Resume, Overdrive-Skip ROM, or Overdrive-Match ROM. Upon completion of an Overdrive-Skip ROM or Overdrive-Match ROM command byte executed at standard speed, the device enters overdrive mode where all subsequent communication occurs at a higher speed. **All 1-Wire data communication is performed least significant bit first.**

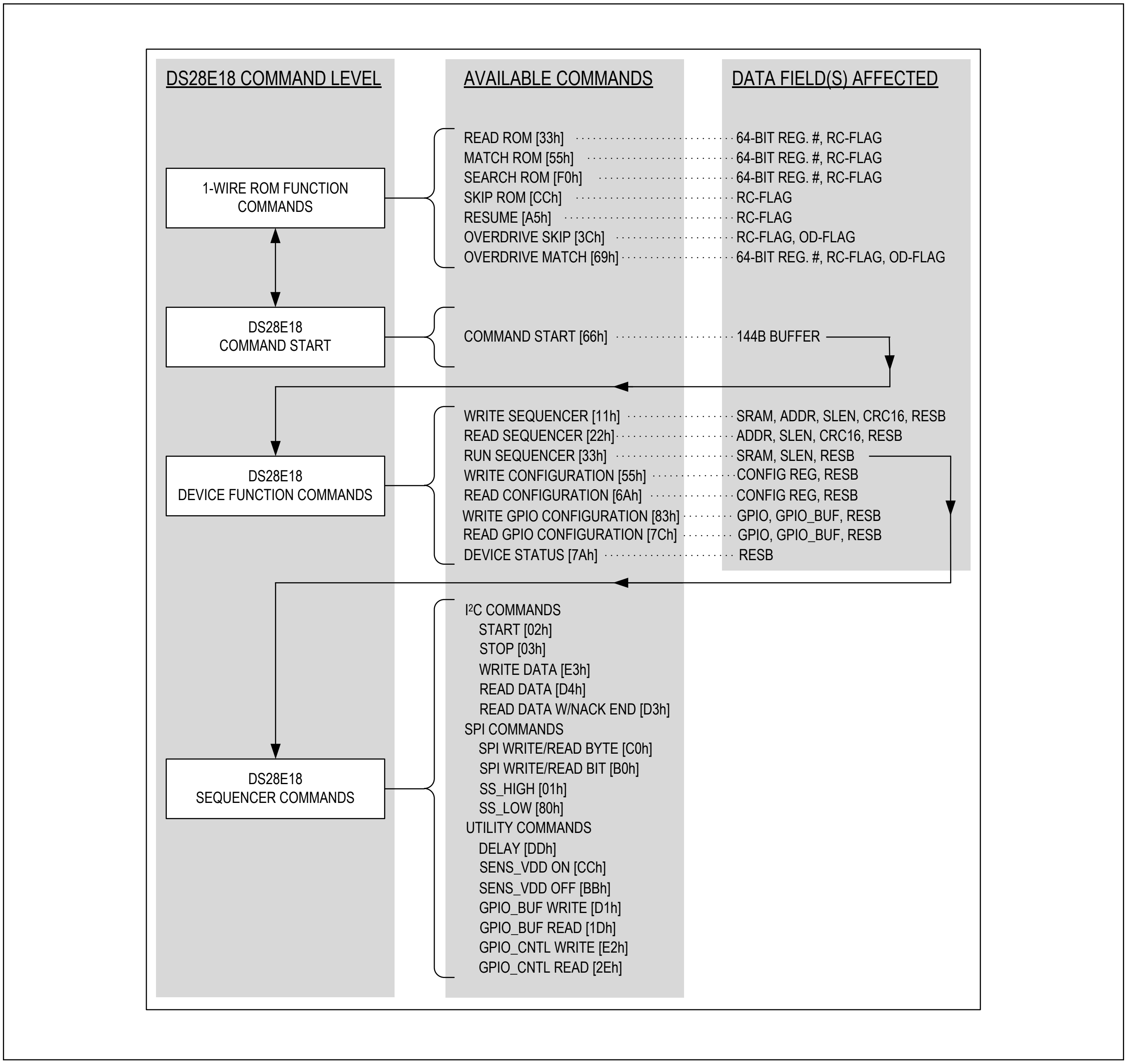


Figure 1. DS28E18 1-Wire Commands Hierarchical Structure

64-Bit ROM ID

Each DS28E18 contains a unique ROM ID that is 64 bits long. The first 8 bits are a 1-Wire family code. The next 48 bits are a unique serial number. The last 8 bits are a cyclic redundancy check (CRC) of the first 56 bits. See Figure 2 for details. The 1-Wire CRC is generated using a polynomial generator consisting of a shift register and XOR gates. The polynomial is $X^8 + X^5 + X^4 + 1$. Additional information about the 1-Wire CRC is available in the *Understanding and Using Cyclic Redundancy Checks with Maxim iButton® Products* application note.

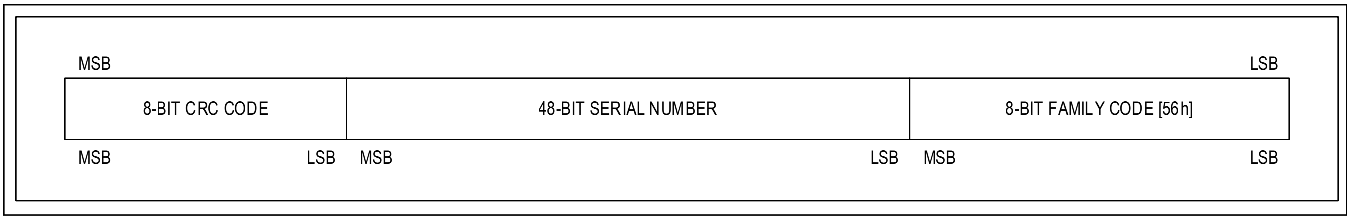


Figure 2. 64-Bit ROM ID

Power-Up ROM ID Serialization

On power-up, the ROM ID value is 56000000000000B2. The uniquely programmed factory value for each DS28E18 needs to be loaded from memory. After power-up, issue a Skip ROM command followed by a Write GPIO Configuration command. This initial command populates the unique device ROM ID, including family code, serialization, and CRC-16. Ignore the command CRC-16 result and the Result byte, as both might be invalid. Next issue a successful Write GPIO Configuration command to configure the GPIO pullup/down states so that the voltage on the GPIO ports is known. See the *Power-Up of GPIO/I²C Pins* section. However, this will not clear the POR status bit. Another successful Device Status command should be issued to receive valid status information and clear the POR status bit.

1-Wire Bus System

The 1-Wire bus is a system that has a single bus master and one or more slaves. In all instances, the DS28E18 is a slave device. The bus master is typically a microcontroller. The discussion of this bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-Wire signaling (signal types and timing). The 1-Wire protocol defines bus transactions in terms of the bus state during specific time slots, which are initiated on the falling edge of sync pulses from the bus master.

Hardware Configuration

The 1-Wire bus has only a single line by definition; it is important that each device on the bus be able to drive it at the appropriate time. To facilitate this, each device attached to the 1-Wire bus must have open-drain or three-state outputs. The 1-Wire port (i.e., IO pin) of the DS28E18 is open drain with an internal circuit equivalent.

A multidrop bus consists of a 1-Wire bus with multiple slaves attached. The DS28E18 supports both a standard and overdrive communication speed of 11kbps (max) and 90kbps (max), respectively. The value of the pullup resistor primarily depends on the network size and load conditions. The DS28E18 requires a pullup resistor of 1kΩ (max) at any speed. Some 1-Wire masters have the pullup resistance (R_{PUP}) built in and others require the addition of R_{PUP} as an external resistor.

The idle state for the 1-Wire bus is high. If for any reason a transaction needs to be suspended, the bus must be left in the idle state if the transaction is to resume. If this does not occur and the bus is left low for more than 16µs in overdrive operation, one or more devices on the bus could be reset.

Transaction Sequence

The protocol for accessing the DS28E18 through the 1-Wire port is as follows:

- Initialization
- ROM function command
- Device function command
- Transaction/data

Initialization

All transactions on the 1-Wire bus begin with an initialization sequence. The initialization sequence consists of a reset pulse transmitted by the bus master followed by presence pulse(s) transmitted by the slave(s). The presence pulse lets the bus master know that the DS28E18 is on the bus and is ready to operate.

1-Wire ROM Function Commands

Once the bus master has detected a presence, it can issue one of the seven ROM function commands that the DS28E18 supports. All ROM function commands are 8 bits long. A list of these commands follows. See the flowcharts in Figure 3 and Figure 4.

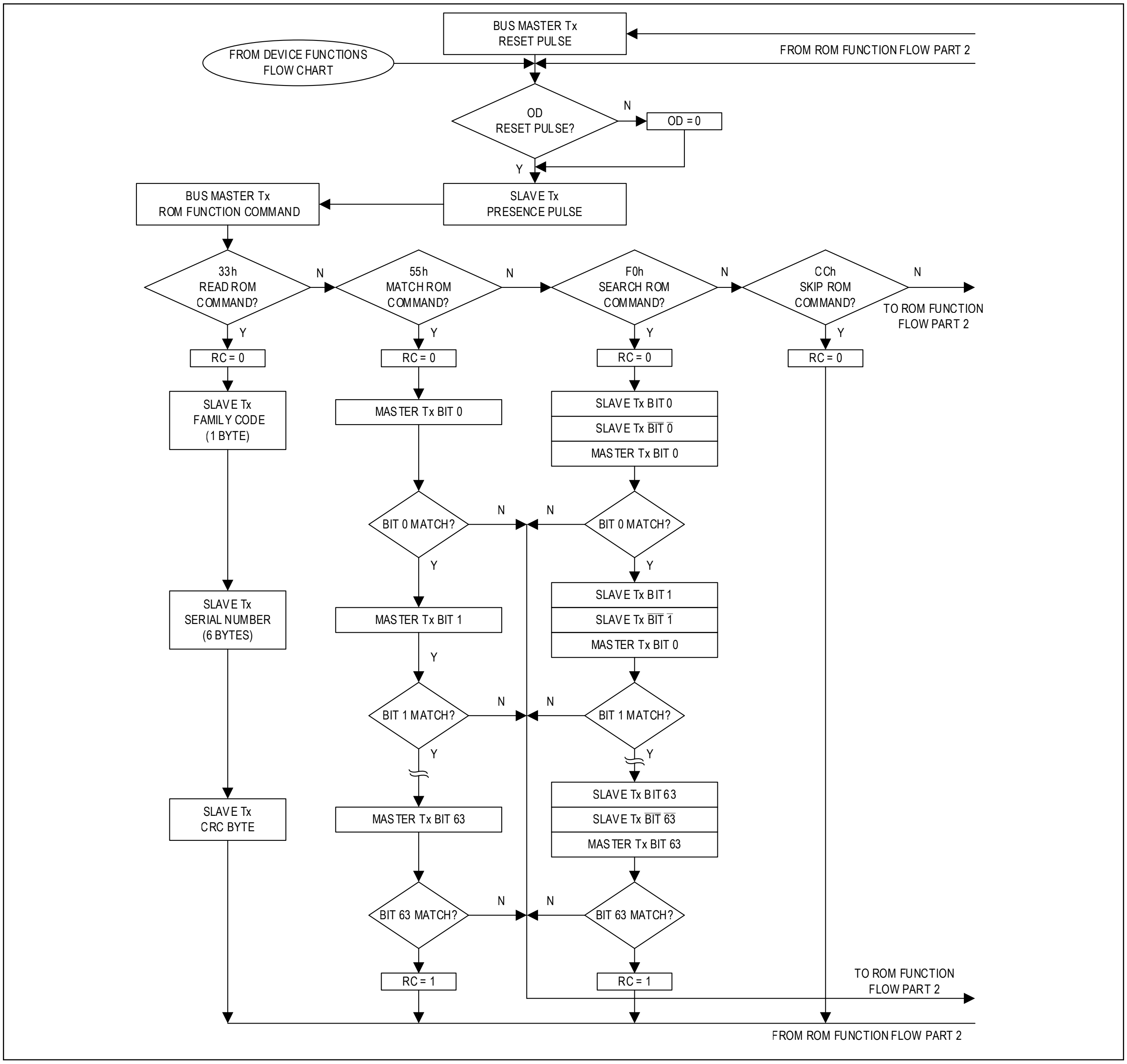


Figure 3. 1-Wire ROM Function Flow Part 1

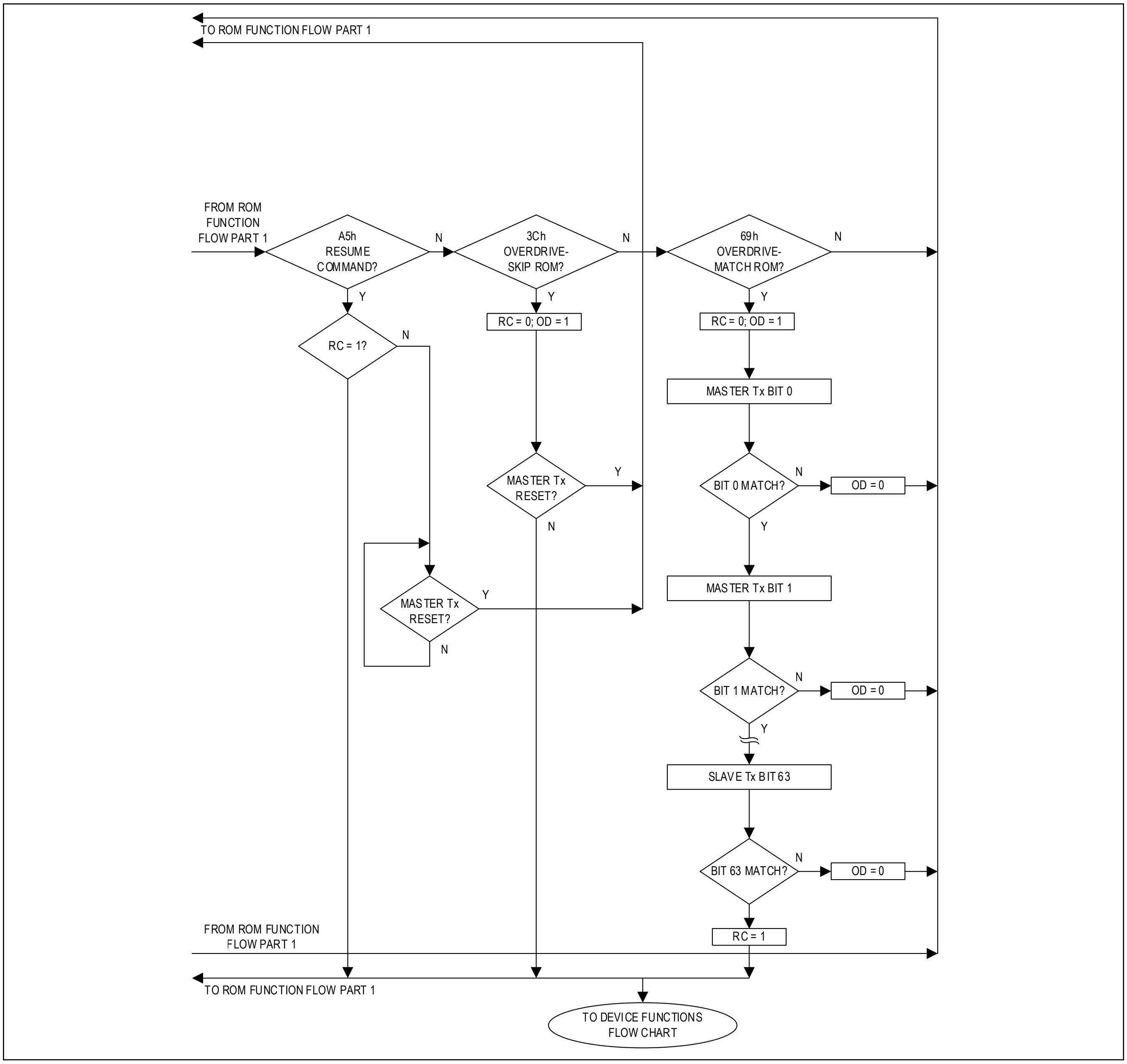


Figure 4. 1-Wire ROM Function Flow Part 2

Search ROM [F0h]

When a system is initially brought up, the bus master might not know the number of devices on the 1-Wire bus or their ROM ID numbers. By taking advantage of the wired-AND property of the bus, the master can use a process of elimination to identify the ID of all slave devices. For each bit in the ID number, starting with the least significant bit, the bus master issues a triplet of time slots. On the first slot, each slave device participating in the search outputs the true value of its ID number bit. On the second slot, each slave device participating in the search outputs the complemented value of its ID number bit. On the third slot, the master writes the true value of the bit to be selected. All slave devices that do not match the bit written by the master stop participating in the search. If both of the read bits are zero, the master knows that slave devices exist with both states of the bit. By choosing which state to write, the bus master branches in the search tree.

After one complete pass, the bus master knows the ROM ID number of a single device. Additional passes identify the ID numbers of the remaining devices. Refer to the *1-Wire Search Algorithm* application note for a detailed discussion, including an example.

Read ROM [33h]

The Read ROM command allows the bus master to read the DS28E18 8-bit family code, unique 48-bit serial number, and 8-bit CRC. This command can only be used if there is a single slave on the bus. If more than one slave is present on the bus, a data collision occurs when all slaves try to transmit at the same time (open drain produces a wired-AND result). The resultant family code and 48-bit serial number result in a mismatch of the CRC.

Match ROM [55h]

The Match ROM command, followed by a 64-bit ROM sequence, allows the bus master to address a specific DS28E18 on a multidrop bus. Only the DS28E18 that exactly matches the 64-bit ROM sequence responds to the subsequent Device command. All other slaves wait for a reset pulse. This command can be used with a single device or multiple devices on the bus.

Skip ROM [CCh]

This command can save time in a single-drop bus system by allowing the bus master to access the Device commands without providing the 64-bit ROM ID. If more than one slave is present on the bus and, for example, a read command is issued following the Skip ROM command, data collision occurs on the bus as multiple slaves transmit simultaneously (open-drain pulldowns produce a wired-AND result).

Resume [A5h]

To maximize the data throughput in a multidrop environment, the Resume command is available. This command checks the status of the RC bit and, if it is set, directly transfers control to the Device commands, similar to a Skip ROM command. The only way to set the RC bit is through successfully executing the Match ROM, Search ROM, or Overdrive-Match ROM command. Once the RC bit is set, the device can repeatedly be accessed through the Resume command. Accessing another device on the bus clears the RC bit, preventing two or more devices from simultaneously responding to the Resume command.

Overdrive-Skip ROM [3Ch]

On a single-drop bus, this command can save time by allowing the bus master to access the device commands without providing the 64-bit ROM ID. Unlike the normal Skip ROM command, the Overdrive-Skip ROM command sets the DS28E18 into the overdrive mode (OD = 1). All communication following this command must occur at overdrive speed until a reset pulse of minimum 480µs duration resets all devices on the bus to standard speed (OD = 0).

When issued on a multidrop bus, this command sets all overdrive-supporting devices into overdrive mode. To subsequently address a specific overdrive-supporting device, a reset pulse at overdrive speed must be issued followed by a Match ROM or Search ROM command sequence. This speeds up the time for the search process. If more than one slave supporting overdrive is present on the bus and the Overdrive-Skip ROM command is followed by a read command, data collision occurs on the bus as multiple slaves transmit simultaneously (open-drain pulldowns produce a wired-AND result).

Overdrive-Match ROM [69h]

The Overdrive-Match ROM command, followed by a 64-bit ROM sequence transmitted at overdrive speed, allows the bus master to address a specific DS28E18 on a multidrop bus and to simultaneously set it in Overdrive mode. Only the DS28E18 that exactly matches the 64-bit ROM sequence responds to the subsequent device command. Slaves already in Overdrive mode from a previous Overdrive-Skip ROM or successful Overdrive-Match ROM command remain in Overdrive mode. All overdrive-capable slaves return to standard speed at the next reset pulse having a minimum duration of 480µs. The Overdrive-Match ROM command can be used with a single device or multiple devices on the bus.

1-Wire Signaling and Timing

The 1-Wire protocol consists of four types of signaling on one line: reset cycle with reset pulse and presence pulse, write-zero, write-one, and read-data. Except for the presence pulse, the 1-Wire master initiates all falling edges. The 1-Wire master can communicate at two speeds: standard and overdrive. While in overdrive mode, the fast timing applies to all wave forms.

Figure 5 shows the initialization sequence required to begin any communication. A reset pulse followed by a presence pulse indicates that a slave is ready to receive data, given the correct ROM and device function command.

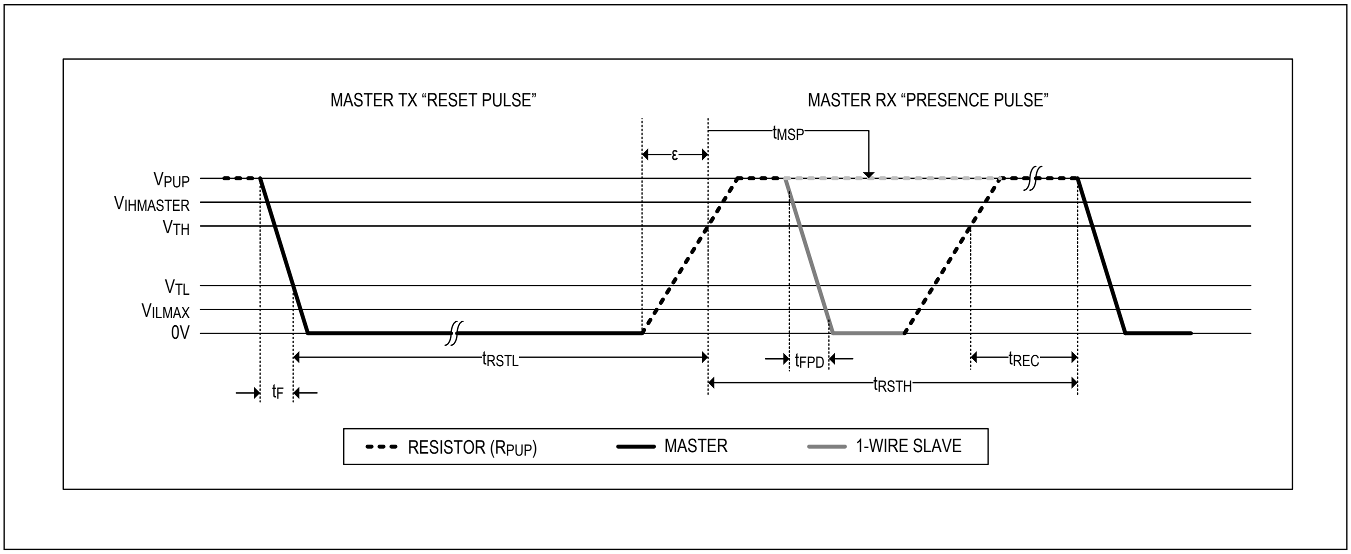


Figure 5. 1-Wire Reset/Presence-Detect Cycle

Read/Write Time Slots

Data communication on the 1-Wire bus takes place in time slots that carry a single bit each. Write time slots transport data from 1-Wire master to a connected slave. Read time slots transfer data from slave to the 1-Wire master. Figure 6 illustrates the definitions of the write and read time slots.

All communication begins with the master pulling the data line low. As the voltage on the 1-Wire line falls below the threshold V_{TL}, the slave starts its internal timing generator that determines when the data line is sampled during a write time slot and how long data is valid during a read time slot.

Master-to-Slave

For a write-one time slot, the voltage on the data line must have crossed the V_{TH} threshold before the write-one low time t_{W1LMAX} is expired. For a write-zero time slot, the voltage on the data line must stay below the V_{TH} threshold until the write-zero low time t_{W0LMIN} is expired. For the most reliable communication, the voltage on the data line should not exceed V_{ILMAX} during the entire t_{W0L} or t_{W1L} window required by the slave. After the V_{TH} threshold has been crossed, the DS28E18 needs a recovery time t_{REC} before it is ready for the next time slot.

Slave-to-Master

A read-data time slot begins like a write-one time slot. The voltage on the data line must remain below V_{TL} until the read low time t_{RL}(read low time) is expired. During the t_{RL} window, when responding with a 0, the slave starts pulling the data line low; its internal timing generator determines when this pulldown ends and the voltage starts rising again. When responding with a 1, the slave does not hold the data line low at all, and the voltage starts rising as soon as t_{RL} is over. Note that the slave t_{RL} during a logic 1 is adequately an approximation of the 1-Wire master t_{W1L} setting.

The slave t_{RL} plus the bus rise time on the near end and the internal timing generator of the slave on the far end define the 1-Wire master sampling window, in which the 1-Wire master performs a read from the data line. After reading from the data line, the 1-Wire master waits until t_{SLOT} is expired. This guarantees sufficient recovery time t_{REC} for the slave to get ready for the next time slot. Note that t_{REC} specified herein applies only to a single slave attached to a 1-Wire line. For multidevice configurations, t_{REC} must be extended to accommodate the additional 1-Wire device input capacitance.

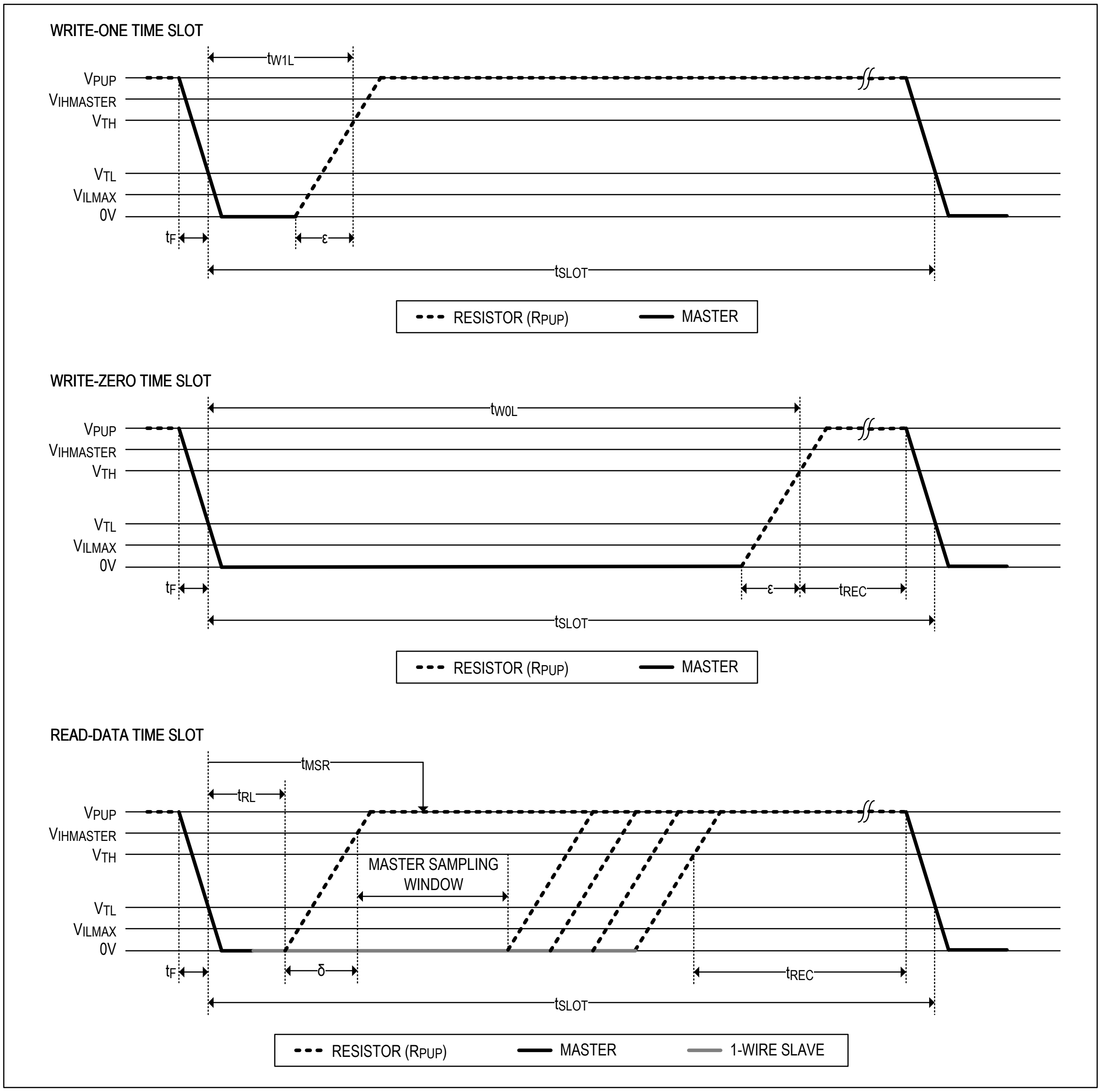


Figure 6. 1-Wire Read/Write Timing Diagrams

Improved Network Behavior

In a 1-Wire environment, line termination is possible only during transients controlled by the bus master (1-Wire driver). 1-Wire networks, therefore, are susceptible to noise of various origins. Depending on the physical size and topology of the network, reflections from end points and branch points can add up or cancel each other to some extent. Such reflections are visible as glitches or ringing on the 1-Wire communication line. Noise coupled onto the 1-Wire line from external sources can also result in signal glitching. A glitch during the rising edge of a time slot can cause a slave device to lose synchronization with the master and, consequently, result in a Search ROM command coming to a dead end or cause a device-specific function command to abort. For better performance in network applications, the DS28E18 uses a 1-Wire front-end that is less sensitive to noise. The IO 1-Wire front-end has hysteresis, and a rising edge hold off delay.

- On the low-to-high transition, if the line rises above V_{TH} but does not go below V_{TL}, the glitch is filtered (Figure 7, case A.)
- The rising edge hold-off delay (nominally 100ns), t_{REH}, filters glitches that go below V_{TL} before t_{REH} has expired (Figure 7, case B.) Effectively the device does not see the initial rise, and the t_{REH} delay resets when the line goes below V_{TL}.
- If the line goes below V_{TL} after t_{REH} has expired the glitch is not filtered and is taken as the beginning of a new time slot (Figure 7, case C.)

Independent of the time slot, the falling edge of the presence pulse has a controlled slew rate to reduce ringing. The falling delay is specified by t_{FPD}.

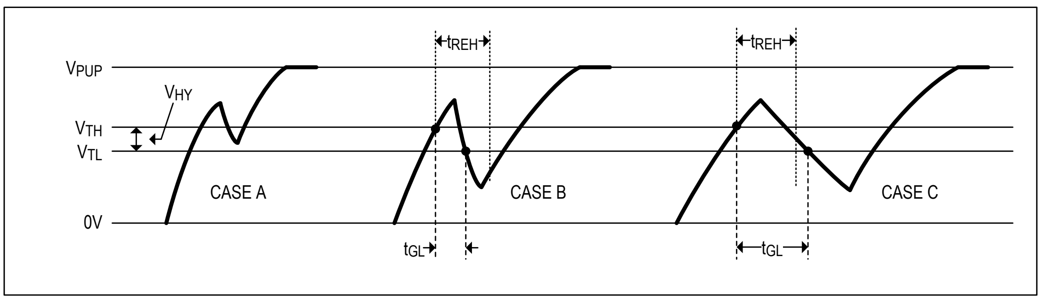


Figure 7. Noise Suppression Scheme

Device Function Commands

After a 1-Wire reset/presence cycle and ROM function command sequence (Figure 3 or Figure 4) is successful, a command start can be accepted and then followed by a device function command. In general, these commands follow the state flow diagram (Figure 8). Within this diagram, the data transfer is verified when writing and reading by a CRC of 16-bit type (CRC-16). The CRC-16 is computed as described in the *Understanding and Using Cyclic Redundancy Checks with Maxim 1-Wire and iButton Products* application note.

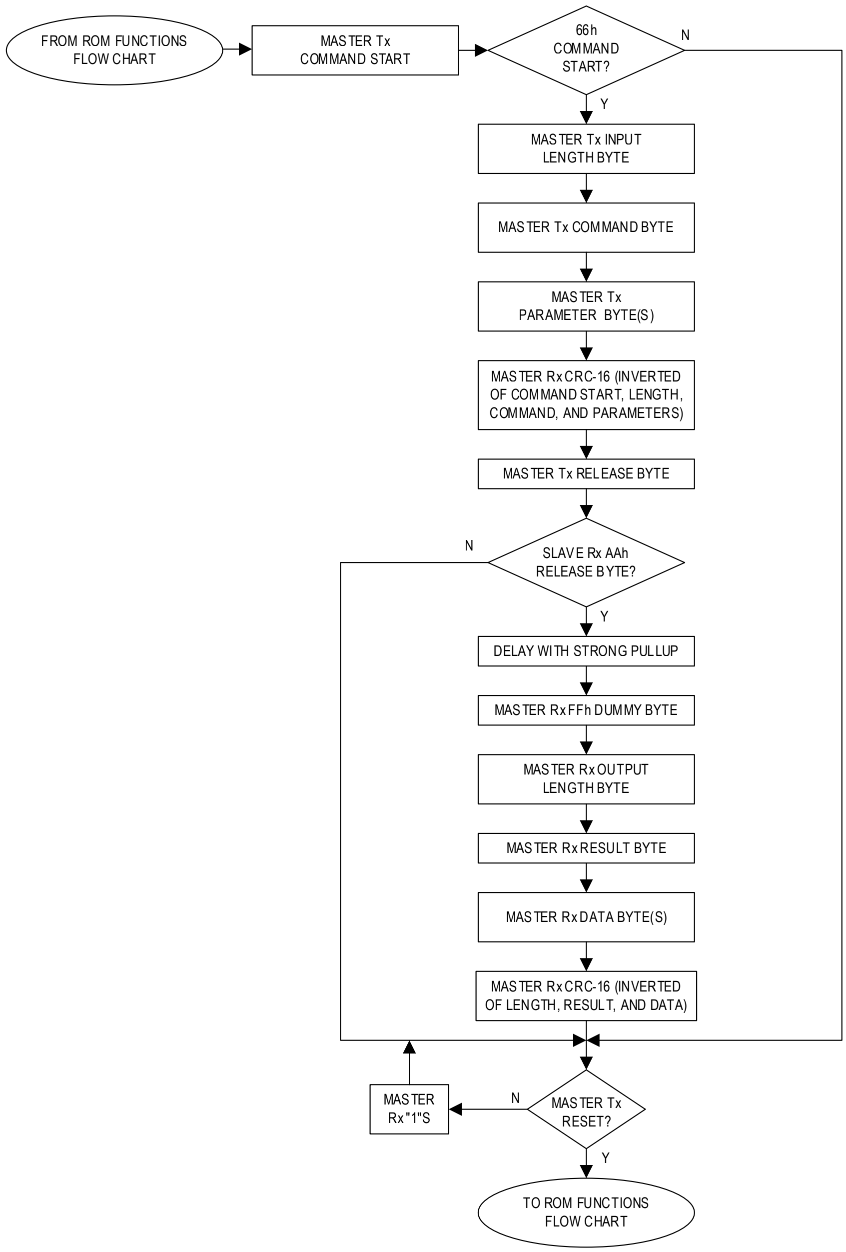


Figure 8. Device Function Flow Chart

Additionally, the subsequent sections will describe each device function command in detail. The device function commands are each 8-bit values and are shown in Table 1.

Table 1. Device Function Commands

DEVICE COMMAND	COMMAND CODE
Write Sequencer	11h
Read Sequencer	22h
Run Sequencer	33h
Write Configuration	55h
Read Configuration	6Ah
Write GPIO Configuration	83h
Read GPIO Configuration	7Ch
Device Status	7Ah

Command Start (66h)

Command Start is used for device function commands. After the command start byte, the next byte transmitted is the length byte. This indicates the length of both the command (i.e., device function command) and parameters. The result of the command is provide in similar format. The command start structure does not require a strong pullup (SPU) until after the release byte. After the release byte, the command is started and a command dependent delay is put into effect with the SPU power being delivered to the 1-Wire bus. The command dependent delay and SPU power is needed to execute device function commands and sequencer commands when applicable.

Table 2. Command Start Description

COMMAND START	
Command Code	66h
Parameter Byte(s)	Length byte followed by command and parameters. The first byte after the length byte is the device function command.
Usage	Process the command and parameters sequence. The command code is followed by a length byte followed by the device function command and parameters. Next, the master receives a two-byte inverted CRC-16 of the command start byte + length byte + command + parameters is sent. If the CRC-16 is correct, the master then sends the release byte (AAh). Once the release byte is received, the command is started. At that time the master must provide strong pullup on the 1-Wire to power the device. The required delay is command dependent with a minimum delay of t_{OP}. After the delay, the master reads a dummy byte for clocking purposes. After the dummy byte, the command result is read, length byte first, followed by a result byte, optional result data, and an inverted CRC-16. If the command is not supported, the response will have a length of 00h followed by the CRC-16 value of FFFFh.
Command Restrictions	None
Device Operation	Verify Release byte is AAh. Start command.
Command Duration	See t_{OP} (command dependent).
Result	Command dependent followed by inverted CRC-16.

Table 3. Generic Command Start Sequence

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (varies with command)
Tx: Command
Tx: Parameters (varies with command)
Rx: CRC-16 (inverted of command start, length, command, and parameters)
Tx: Release Byte (AAh)
<SPU Delay, command dependent>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (varies with command)
Rx: Result Byte (varies with command)
Rx: Result Bata (varies with command)
Rx: CRC-16 (inverted of length byte, result byte, and result data)
Reset

Write Sequencer Command (11h)

Table 4. Write Sequencer Description

WRITE SEQUENCER	
Command Code	11h
Parameter Byte(s)	ADDR_LO, ADDR_HI, data array to write
Usage	This command writes up to 128-bytes to the 512-byte command sequencer SRAM starting at the specified address.
Command Restrictions	The sum of the specified address and length must be within the command sequencer valid address range, that is to not exceed 512 bytes in the sequencer memory. If 512 bytes is exceeded, nothing will be written.
Device Operation	Write up to the maximum buffer length of 128 bytes of data to the command sequencer SRAM. The command processor sets the result byte after verifying the parameters and writing the data.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success

Table 5. Write Sequencer Parameter: ADDR_LO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ADDR_LO							

ADDR_LO: (Bits 7:0)

00h-FFh: The lower 8 bits of the target write address in the command sequencer SRAM.

Table 6. Write Sequencer Parameter ADDR_HI

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU							ADDR_HI

RFU (Bits 7:1): RFU

0000000b-1111111b: Reserved for future use.

ADDR_HI: (Bit 0): Sequencer Address High Bit

0b-1b: The most significant bit of the target write address in the command sequencer SRAM.

Table 7. Write Sequencer Parameter: Data Array

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DATA							

DATA: (Bits 7:0)

00h-FFh: Byte to write to the command sequencer SRAM starting at the target write address (*ADDR_HI:ADDR_LO*).
DATA is a variable length array of bytes, *DATA[n]*, that is written to the command sequencer SRAM. The length of the array, *n,* is determined from the length parameter as part of the 1-Wire Start command. A minimum of one byte of data is required for the Write Sequencer command. The array, *DATA[n]*, is written to the command sequencer SRAM in the same order it is transmitted.
Note: *The 1-Wire protocol transmits the data least significant bit to most significant bit. The DS28E18 1-Wire slave interface receives the 1-Wire data and writes it in little endian form to the internal command buffer first and then transfers data during the command duration to the command sequencer SRAM.*

Table 8. Write Sequencer Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (4 to 131) Command + ADDR_LO + ADDR_HI + DATA[*n*]
Tx: Command 11h (Write Sequencer
Tx: Parameter (ADDR_LO)
Tx: Parameter (ADDR_HI)
Tx: Data (*n* = 1 to 128 bytes)
Rx: CRC-16 (inverted of command start, length, command, parameters, data array)
Tx: Release Byte (AAh)
<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (1)
Rx: Result Byte
Rx: CRC-16 (inverted of length and result byte)
Reset

Read Sequencer Command (22h)

Table 9. Read Sequencer Description

READ SEQUENCER	
Command Code	22h
Parameter Byte(s)	See below
Usage	Read up to 128 bytes from the 512-byte command sequencer SRAM with the Read Sequencer command.
Command Restrictions	SLEN + ADDR must be less than or equal to 512. 0 ≤ SLEN ≤ 127 ***Note:*** *If SLEN equals 0, the length of the read is 128 bytes.*
Device Operation	Validates parameters. Reads data from the command sequencer SRAM. Sets the Result byte. Returns data read.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success

Table 10. Read Sequencer Parameter ADDR_LO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ADDR_LO							

ADDR_LO: (Bits 7:0)

00h-FFh: The lower 8 bits of the target read address in the command sequencer SRAM.

Table 11. Read Sequencer Parameter Byte 2: SLEN:ADDR_HI

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SLEN							ADDR_HI

SLEN: (Bits 7:1)

00h-7Fh: The number of bytes to read from the command sequencer SRAM.
Note: *Setting the SLEN field to 0 reads 128 bytes, the maximum number.*

ADDR_HI: (Bit 0)

0b-1b: The most significant bit of the target read address in the command sequencer SRAM. Combine the ADDR_HI bit with the ADDR_LO field to set the starting read address for the Read Sequencer command.

Table 12. Read Sequencer Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (3)
Tx: Command 22h (Read Sequencer)
Tx: Parameter (ADDR_LO)
Tx: Parameter (SLEN:ADDR_HI)
Rx: CRC-16 (inverted of command start, length, command, and parameters)
Tx: Release Byte (AAh)

Table 12. Read Sequencer Command Transfer (continued)

<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (1 ≤ *m* ≤ 129 max)
Rx: Result Byte
Rx: Data Array up to 128 bytes (if *m* is > 1, receive *(m-1)* bytes)
Rx: CRC-16 (inverted of length, result byte, and data array bytes)
Reset

Run Sequencer Command (33h)

Table 13. Run Sequencer Description

RUN SEQUENCER	
Command Code	33h
Parameter Byte(s)	ADDR_LO, SLEN_LO:ADDR_HI, and RFU:SLEN_HI. Details below.
Usage	The Run Sequencer command is used to execute properly formed sequencer packets stored in the 512-byte command sequencer SRAM. The execution starting address is a 9-bit field and is the combination of *ADDR_HI:ADDR_LO*. The number of sequencer bytes to execute is also a 9-bit field and is a combination of *SLEN_HI:SLEN_LO* (Sequencer Length). Setting SLEN_HI:SLEN_LO to 0 results in an execution length of 512 bytes, the maximum value for execution. If SLEN_HI:SLEN_LO is 0, the ADDR_HI:ADDR_LO field must also be 0 or an error condition occurs, and result byte is set to 77h.
Command Restrictions	The maximum number of bytes for a sequencer execution is 512. The address and number of bytes to execute must be within the command sequencer SRAM space. If (ADDR_HI:ADDR_LO) + (SLEN_HI:SLEN_LO) > 512 an error condition occurs, and the Run Sequencer command is aborted prior to execution of any sequencer SRAM commands. If SLEN_HI:SLEN_LO is 0, ADDR_HI:ADDR_LO must be 0 or an error condition is returned. ***Important:*** *The starting address and sequencer length value should be set as to always encompass complete and valid sequencer packets. Additionally, the Device Status POR bit must not be set, otherwise sequencer packets will not be executed and the Result byte returned will be 44h.*
Device Operation	The command processor executes the specified sequencer packets after the 1-Wire release byte is received from the host. After this 1-Wire release byte, SPU must be active to supply the power necessary for the sequencer. When the command duration has expired, SPU is to become inactive at the beginning of the received dummy byte. The command processor sets the result byte. If the result byte is 88h, indicating a NACK occurred during an I^2C transaction, the command processor sets the S*NACK_LO* byte and S*NACK_HI* bit and transfers them with the result byte.
Command Duration	t_{OP} + total sequencer communication time (See Table 44, Table 45, and Table 46 for time per command) ***Important:*** *These three tables include the communication time for each sequencer command at a particular speed. By summing the time for each command in the sequence, the total communication time can be determined. The strong pullup must remain active for the full duration,* t_{OP} *+ total sequencer communication time.*
Result Byte	44h: POR occurred resulting in the command sequencer memory being set to zero. 55h: Execution Error (Sequencer command packet or packets incorrectly formed) 77h: Invalid input or parameter 88h: NACK occurred on a data byte (I^2C only) AAh: Success

Table 14. Sequencer Parameter ADDR_LO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ADDR_LO							

ADDR_LO: (Bits 7:0)

00h-FFh: The lower 8 bits of the starting execution address in the command sequencer SRAM.

Table 15. Sequencer Parameter SLEN_LO:ADDR_HI

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SLEN_LO							ADDR_HI

SLEN_LO: (Bits 7:1)

00h-7Fh: The lower 7 bits of the number of bytes to execute in the command sequencer SRAM. Combining SLEN_HI:SLEN_LO (sequencer length) is a 9-bit field. Setting SLEN_HI:SLEN_LO to 0 indicates 512 bytes for the number of bytes to execute.
Note: *If set to zero in conjunction with SLEN_HI being zero, then the sequencer length is 512.*

ADDR_HI: (Bit 0)

0b-1b: The most significant bit of the target read address in the command sequencer SRAM.

Table 16. Sequencer Parameter SLEN_HI

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU						SLEN_HI	

RFU (Bits 7:2)

00b-11b: Reserved for future use.

SLEN_HI: (Bits 1:0)

00b-11b: The upper 2 bits of the number of bytes to execute in the command sequencer SRAM.

Table 17. Sequencer Return Byte Slave NACK Status Byte Low

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SNACK_LO							

SNACK_LO (Bits 7:0): Slave NACK Address Offset Low

00h-FFh: If the Result byte is 88h, a slave NACK occurred during the I^2C sequence. The SNACK_LO byte is returned as part of the run sequencer result. This field must be combined with the SNACK_HI byte to determine the full address offset of the command.
Note: *This parameter is only returned if the DS28E18 is configured for I^2C operation and a NACK occurred during the run sequencer operation.*

Table 18. Sequencer Return Byte Slave NACK Status Byte High

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU							SNACK_HI

RFU (Bits 7:1)

00h-7Fh: Reserved for future use.

SNACK_HI (Bit 0): Slave NACK Address Offset High

0b-1b: If the Result byte is 88h, a slave NACK occurred during the I^2C sequence. The SNACK_HI bit is returned as part of the run sequencer result. SNACK_HI bit is the most significant bit of the first I^2C write byte that was not acknowledged. The full offset of the I^2C Write command that was not acknowledged is the combination of SNACK_HI:SNACK_LO. This field must be combined with the SNACK_HI byte to determine the full address offset of the command.
Notes:
If SNACK_HI:SNACK_LO is 0, the offset address is 512.
This parameter is only returned if the DS28E18 is configured for I^2C operation and a NACK occurred during the run sequencer operation.

Table 19. Slave NACK Addressing

SNACK_HI	SNACK_LO	NACK OFFSET
0b	00000000b	512
0b	00000001b	1
0b	00000010b	2
0b	00000011b	3
0b	00000100b	4

Table 19. Slave NACK Addressing (continued)

SNACK_HI	SNACK_LO	NACK OFFSET
:	:	:
1b	00000000b	256
1b	00000001b	257
:	…	:
1b	11111111b	511

Table 20. Run Sequencer Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (4)
Tx: Command 33h (run sequencer)
Tx: Parameter (ADDR_LO)
Tx: Parameter (SLEN_LO:ADDR_HI)
Tx: Parameter (SLEN_HI)
Rx: CRC-16 (inverted of command start, length, command, and parameters)
Tx: Release Byte (AAh)
<See command duration>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (1 or 3)
Rx: Result Byte
If Result Byte == 88h
Rx: Slave NACK Byte Low (SNACK_LO)
Rx: SNACK Byte High (SNACK_HI)
Rx: CRC-16 (inverted of length, result byte, and NACK bytes, if present)
Reset

Device Configuration and Status Commands

Use the following device commands to configure the general-purpose I/O, I²C, or SPI interface of the DS28E18.

Write Configuration Command (55h)

Table 21. Write Configuration Description

WRITE CONFIGURATION	
Command Code	55h
Parameter Byte(s)	See below.
Usage	Set up the I²C or SPI interface for the device.
Command Restrictions	Write only command.
Device Operation	Validates the data packet and CRC. Validates the configuration register. Configures the I²C/SPI interface. Sets the Result byte.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success

Table 22. Configuration Register

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU		SPI_MODE		PROT	INACK	SPD	

RFU (Bit 7:6)

00b-11b: Reserved for future use.

SPI_MODE (Bits 5:4): SPI Mode Selection

00b: ***SPI Mode 0:*** The rising edge clocks data from MISO into the DS28E18 and data from MOSI into the SPI slave. The DS28E18 updates its MOSI pin at the falling edge. Clock is active high and idle low (power-on default).

01b: Reserved for future use (invalid).

10b: Reserved for future use (invalid).

11b: ***SPI Mode 3:*** The rising edge clocks data from MISO into the DS28E18 and data from MOSI into the SPI slave. The DS28E18 updates its MOSI pin at the falling edge. Clock is active low and idle high.

PROT (Bit 3): Protocol Selection

0b: I²C mode (default)

1b: SPI mode

INACK (Bit 2): Ignore NACK

0b: If a NACK occurs, stop processing the sequencer packets and set the Write Status byte value.

1b: If a NACK occurs, populate the first I²C write NACK received in the Write Status byte but continue processing the remaining sequencer packets. If another NACK occurs, the Write Status byte will still just show the address of the first NACK.

SPD (Bits 1:0): Speed

00b: I²C/SPI speed set to 100kHz max

01b: I²C/SPI speed set to 400kHz max (power-on default)

10b: I²C/SPI speed set to 1MHz max

11b: SPI speed set to 2.3MHz max

Note: In SPI mode, the SPD bits only affect speed for the SPI WRITE/READ BYTE sequencer command. The speed for the SPI WRITE/READ BIT sequencer command is variable up to a maximum of 130kHz.

Table 23. Write Configuration Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (2)
Tx: Command 55h (write configuration)
Tx: Configuration Byte
Rx: CRC-16 (inverted of command start, length, command, and configuration byte)
Tx: Release Byte (AAh)
<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (1)
Rx: Result Byte
Rx: CRC-16 (inverted of length and result byte)
Reset

Read Configuration Command (6Ah)

Table 24. Read Configuration Description

READ CONFIGURATION	
Command Code	6Ah
Parameter Byte(s)	See below.
Usage	Read the DS28E18 configuration register.
Command Restrictions	Read only command.
Device Operation	Reads the configuration register. Sets the Result byte.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success

Table 25. Configuration Register Return Byte

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU		SPI_MODE		PROT	INACK	SPD	

RFU (Bit 7:6)

00b-11b: Reserved for future use.

SPI_MODE (Bits 5:4): SPI Mode Selection

00b: ***SPI Mode 0:*** The rising edge clocks data from MISO into the DS28E18 and data from MOSI into the SPI slave. The DS28E18 updates its MOSI pin at the falling edge. Clock is active high and idle low (power-on default).

01b: Reserved for future use (invalid).

10b: Reserved for future use (invalid).

11b: ***SPI Mode 3:*** The rising edge clocks data from MISO into the DS28E18 and data from MOSI into the SPI slave. The DS28E18 updates its MOSI pin at the falling edge. Clock is active low and idle high.

PROT (Bit 3): Protocol Selection

0b: I²C mode (default)

1b: SPI mode

INACK (Bit 2): Ignore NACK

0b: If a NACK occurs, stop processing the sequencer packets and set the Write Status byte value.

1b: If a NACK occurs, populate the first I²C write NACK received in the Write Status byte, but continue processing the remaining sequencer packets. If another NACK occurs, the Write Status byte will still just show the address of the first NACK.

SPD (Bits 1:0): Speed

00b: I²C/SPI speed set to 100kHz max

01b: I²C/SPI speed set to 400kHz max (power-on default)

10b: I²C/SPI speed set to 1MHz max

11b: SPI speed set to 2.3MHz max

Table 26. Read Configuration Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (1)
Tx: Command 6Ah (read configuration)
Rx: CRC-16 (inverted of command start, length, command)
Tx: Release Byte (AAh)
<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (2)
Rx: Result Byte
Rx: Configuration Register (1)
Rx: CRC-16 (inverted of length and result byte)
Reset

Write GPIO Configuration (83h)

Table 27. Write GPIO Configuration Description

WRITE GPIO CONFIGURATION	
Command Code	83h
Parameter Byte(s)	See below.
Usage	Write the configuration of GPIOA/GPIOB and SDA/SCL.
Command Restrictions	Module must equal 3, target must equal either 0Bh or 0Ch.
Device Operation	Writes the GPIO configuration information and returns the result. Sets the Result byte.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success

Table 28. Target Configuration Register - CFG_REG_TARGET

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OFFSET							

OFFSET (Bit 7:0): GPIO Register Address Offset

0Bh: Sets access to the GPIO control register. Adhere to the GPIO_CTRL_HI/GPIO_CTRL_LO register descriptions.

0Ch: Sets access to the GPIO buffer register. Adhere to the GPIO_BUF_HI/GPIO_BUF_LO register descriptions.

Table 29. Target Configuration Register Module - CFG_REG_MOD

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
GPIO_MOD							

GPIO_MOD (Bits 7:0): GPIO Register Module

03h: GPIO register module number sets access to the GPIO peripheral.

Table 30. GPIO Control Register High Byte - GPIO_CTRL_HI

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PS				PW			

Note: See Table 32 for mapping to pin names.

PS (Bit [n]): Pull Strong Setting

0h-Fh: See Table 31.

PW (Bit [n]): Pull Weak Setting

0h-Fh: See Table 31.

Table 31. Pullup Selection

PS[n] BIT*	PW[n] BIT*	SETTING DESCRIPTION
0	0	External pullup resistor, open drain
0	1	25kΩ internal pullup resistor, open drain
1	0	2.7kΩ internal pullup resistor, open drain
1	1	Hard drive based on the value in DO_n bit.

*The bit positions [n] are 0, 1, 2, and 3. Each [n] of PS must equal [n] of PW for the table.

Table 32. Bit Position [n] Mapping to Pin Names

NIBBLE BIT POSITION [n]	PIN NAMES
3	SDA/MOSI
2	GPIOB/MISO
1	SCL/SCLK
0	GPIOA/SS#

Table 33. GPIO Control Register Low Byte - GPIO_CTRL_LO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
PDS				DO			

Note: See Table 32 for mapping to pin names.

PDS (Bit [n]): Pulldown Slew Setting

0b: No pulldown slew

1b: Pulldown slew, t_F = 300ns (typ)

DO (Bit [n]): Output Data Setting

0b: Output low

1b: Output high or release line depending on pullup selected

Table 34. GPIO Buffer Register Configuration Setting High Byte - GPIO_BUF_HI

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU				BUFIZ			

Note: See Table 32 for mapping to pin names.

RFU (Bits 7:4): Reserved for future use. Set to 0.

BUFIZ (Bit [n]): Input Buffer Enable

0b: Input buffer is isolated (high-Z).

1b: Input buffer is enabled.

Note: The BUFIZ bits control the input buffers outside of the device function command, and should be kept at default, 0. The input buffers are forced on during device function commands. Therefore, make sure to turn on a pullup/pulldown or provide an external pullup per Table 31. This avoids any chance of excess crowbar current for the condition of when the associated GPIO or SDA/SCL is at mid-rail or floating. See the *Power-Up of GPIO/I²C Pins* section for more details.

Table 35. GPIO Buffer Register Configuration Setting Low Byte (Write) - GPIO_BUF_LO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU				X	X	X	X

Note: X = Don't care.

RFU (Bits 7:4): Reserved for future use.

X (Bits 3:0): Don't care. These bits are read only for the buffer, and the write will be ignored.

Table 36. Write GPIO Configuration Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)

Table 36. Write GPIO Configuration Command Transfer (continued)

Tx: Length Byte (5)
Tx: Command 83h (write GPIO configuration)
Tx: Parameter (CFG_REG_TARGET)
Tx: Parameter (CFG_REG_MOD)
Tx: Parameter (GPIO_CTRL_HI or GPIO_BUF_HI)
Tx: Parameter (GPIO_CTRL_LO or GPIO_BUF_LO)
Rx: CRC-16 (inverted of command start, length, command, and all parameters)
Tx: Release Byte (AAh)
<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (1)
Rx: Result Byte
Rx: CRC-16 (inverted of length and result byte)
Reset

Read GPIO Configuration (7Ch)

READ GPIO CONFIGURATION	
Command Code	7Ch
Parameter Byte(s)	See below.
Usage	Read the DS28E18 GPIO configuration register.
Command Restrictions	Reads the GPIO configuration bytes only. Module must equal 3, target must equal either 0Bh or 0Ch.
Device Operation	Reads the GPIO configuration information and returns it to the host. Sets the Result byte.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success

Target Configuration Register – CFG_REG_TARGET

See Table 28 for byte description.

Target Configuration Register Module – CFG_REG_MOD

See Table 29 for byte description.

GPIO Control Register High Byte – GPIO_CTRL_HI

See Table 30 for bit descriptions.

GPIO Control Register Low Byte – GPIO_CTRL_LO

See Table 33 for bit descriptions.

GPIO Buffer Register Configuration Setting High Byte – GPIO_BUF_HI

See Table 34 for bit descriptions.

Table 37. GPIO Buffer Register Configuration Setting Low Byte – GPIO_BUF_LO

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU				BUF			

Note: See Table 32 for mapping to pin names.

RFU (Bits 7:4): Reserved for future use.

BUF (Bit [n]): Input Buffer

0b: Read input buffer value is 0.

1b: Read input buffer value is 1.

Table 38. Read GPIO Configuration Command Transfer

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (3)
Tx: Command 7Ch (read GPIO configuration)
Tx: Parameter (CFG_REG_TARGET)
Tx: Parameter (CFG_REG_MOD)
Rx: CRC-16 (inverted of command start, length, command and all parameters)
Tx: Release Byte (AAh)

Table 38. Read GPIO Configuration Command Transfer (continued)

<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation)
Rx: Length Byte (3)
Rx: Result Byte
Rx: Configuration High Byte read from GPIO_CTRL_HI or GPIO_BUF_HI
Rx: Configuration Low Byte read from GPIO_CTRL_LO or GPIO_BUF_LO
Rx: CRC-16 (inverted of length, result byte, and data bytes)
Reset

Device Status Command (7Ah)

Table 39. Device Status Description

DEVICE STATUS	
Command Code	7Ah
Parameter Byte(s)	See below.
Usage	Read the device status information.
Command Restrictions	Read device status, version byte, and MANID bytes.
Device Operation	Reads the device configuration information. Returns the Device Status byte and POR bit is cleared. Returns the version. Returns the MANID bytes to the host. Sets the Result byte.
Command Duration	t_{OP}
Result Byte	77h: Invalid input or parameter AAh: Success (If previously set, POR bit cleared)

Table 40. Device Status Byte

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU						POR	RFU

RFU (Bit 7:2)

0b000000: Reserved for future use.

POR (Bit 1): Power-on Reset Bit

0b0: No POR has occurred.

0b1: A POR has occurred. Buffer and SRAM contents may not be valid.

RFU (Bit 0)

0b0: Reserved for future use.

Table 41. Version Byte

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DVER							

DVER: (Bits 7:0): Device Version

0x00: Device version as set during manufacturing.

Table 42. MANID: (Bits 15:0): Manufacturing ID

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MANID[1]								MANID[0]							

MANID[1] (Bits 15:8): Manufacturing Identification MSB

0x00: Manufacturer identification number, most significant byte, set during manufacturing.

MANID[0] (Bits 7:0): Manufacturing Identification LSB

0x00: Manufacturer identification number, least significant byte, set during manufacturing.

Table 43. Device Status Command Transfer

Reset

Table 43. Device Status Command Transfer (continued)

Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (1)
Tx: Command 7Ah (device status)
Rx: CRC-16 (inverted of command start, length, command)
Tx: Release Byte (AAh)
<Delay t_{OP}>
Rx: Dummy Byte (not used in calculation)
Rx: Length Byte (5)
Rx: Result Byte
Rx: Device Status Byte
Rx: Version Byte
Rx: MANID[0]
Rx: MANID[1]
Rx: CRC-16 (inverted of length, result byte, status byte, version, and MANID bytes)
Reset

Sequencer Commands

This section describes the 8-bit sequencer commands that are used to form packets in the sequencer SRAM memory. The SRAM reserves 512 bytes of the SRAM memory for storage of user-defined command sequences. The maximum size of a Sequencer Run command is 512 bytes. The sequencer does not allow execution beyond the sequencer SRAM and if the starting address plus the sequencer length (SLEN) results in a value greater than 512, an error is returned. Table 44 lists the I²C interface and GPIO commands, Table 45 lists the SPI interface commands and Table 46 lists the utility commands. These three tables include the communication time for each sequencer command at a particular speed. By summing the time for each command in the sequence, the total communication time can be determined. The strong pullup must remain active for the full duration t_{OP} + total sequencer communication time.

Table 44. I²C Interface Commands

COMMAND	COMMAND VALUE	DETAILS	SEQUENCER COMMUNICATION TIME (µs)		
			SPD = 00b	SPD = 01b	SPD = 10b
Start	02h	Send an I²C Start	33	12	8
Stop	03h	Send an I²C Stop	33	12	8
Write Data	E3h	I²C Write Byte(s)	136 (Per byte)	45 (Per byte)	25 (Per byte)
Read Data	D4h	I²C Read Byte(s)	135 (Per byte)	44 (Per byte)	24 (Per byte)
Read Data w/ NACK End	D3h	I²C Read Data Byte with NACK	135 (Per byte)	44 (Per byte)	24 (Per byte)

Table 45. SPI Interface Commands

COMMAND	COMMAND VALUE	DETAILS	SEQUENCER COMMUNICATION TIME(µs)			
			SPD = 00b	SPD = 01b	SPD = 10b	SPD = 11b
SPI Write/ Read Byte	C0h	Write or Read a full byte from the SPI interface	123 (Per byte)	42 (Per byte)	25 (Per byte)	17 (Per byte)
SPI Write/ Read Bit	B0h	Write or Read from 1 to 64 bits from the SPI Interface	26 (Per single bit; not affected by SPD)			
SS_HIGH	01h	Sets the slave select output high	35	14	10	8
SS_LOW	80h	Sets the slave select output low	35	15	10	8

Table 46. Utility Commands

COMMAND	COMMAND VALUE	DETAILS	SEQUENCER COMMUNICATION TIME (µs)
Delay	DDh	Perform a delay between sequencer commands from 1ms to 32s.	1248 (1ms setting)
SENS_VDD On	CCh	Turn the SENS_VDD power output on.	6
SENS_VDD Off	BBh	Turn the SENS_VDD power output off.	6
GPIO_BUF Write	D1h	Write a value to the GPIO_BUF register.	8

Table 46. Utility Commands (continued)

COMMAND	COMMAND VALUE	DETAILS	SEQUENCER COMMUNICATION TIME (µs)
GPIO_BUF Read	1Dh	Read the GPIO_BUF register.	8
GPIO_CTRL Write	E2h	Write a configuration to the GPIO_CTRL register.	9
GPIO_CTRL Read	2Eh	Read the current configuration from the GPIO_CTRL register	10

Sequencer commands are grouped into three categories; I²C interface, SPI interface, and utility commands. The I²C interface commands exercise the I²C bus; and the SPI interface commands exercise the SPI bus. The utility commands serve to provide time for the I²C/SPI sensors to process instructions or extract power from 1-Wire for power delivery to the I²C/SPI sensors.

Construct packets for the target interface in the command sequencer SRAM using the Write Sequencer device command. Once the command packets are written to the command sequencer SRAM, they remain in SRAM until a device reset or overwritten. The command packets typically contain I²C/SPI write data bytes to be written or an array of data bytes each set to FFh as a way to preserve memory that will later be overwritten in SRAM with received I²C/SPI read data. Consider that it may be important to add a delay or output power for a I²C/SPI slave when processing write/read data (e.g., during a I²C temperature conversion, memory write, etc.). This is accomplished using utility commands that can be inserted into the sequencer.

Execution of the command packets stored in the command sequencer SRAM is initiated using the Run Sequencer device command. All sequencer commands and parameter bytes that form a packet (e.g., write length, write data, read array, etc.) are processed. If read data bytes are incoming during the processing of the Run Sequencer command, they will overwrite the read array (previously set to FFh for each byte) in SRAM with the new received read data.

When data is read over the I²C or SPI interface during the Run Sequencer command, such as collecting sensor data, the host should retrieve the read array stored in SRAM. The data stored in the SRAM can be extracted by calling a Read Sequencer device command. This addressable command retrieves up to 128 bytes of read array at a time in the SRAM over the 1-Wire. Since the *Read Sequencer* command is addressable, it can be called as much as is needed to get all of the read array bytes that contain the I²C/SPI received data. After collecting the read array, it may make sense for the specific application to rewrite FFh in the read arrays of the SRAM again by using the Write Sequencer command. This is just an added precaution to be sure that the next time the Read Sequencer command is called, the read array has truly been updated from a Run Sequencer command with expected data other than FFh for each byte from the I²C/SPI interface.

I²C Sequencer Interface Commands

Table 47 shows abbreviations for the I²C communication types used in the I²C sequencer interface commands descriptions. Each abbreviation translates to a specific I²C communication type.

See Table 48 for the color coding of the I²C communication direction shown in each command's expected I²C transaction diagram.

Table 47. I²C Character Legend

LEGEND	I²C EQUIVALENT
S	Start
P	Stop
ACK	Acknowledged
NACK	Not Acknowledged

Table 48. I²C Data Direction Color Key

I²C COLOR CODES
Master-to-Slave
Slave-to-Master

I²C Start Command

Table 49. I²C Start Command

I²C START COMMAND	
Sequencer Command	02h
Typical Usage	Use this to perform an I²C Start condition.
I²C Features	Start or Repeated Start.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command 02h

Expected I²C Transaction

S

I²C Stop Command

Table 50. I²C Stop Command

I²C STOP COMMAND	
Sequencer Command	03h
Typical Usage	Use this to perform an I²C Stop condition.
I²C Features	Stop.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command 03h

Expected I²C Transaction

P

I²C Write Data Command

Table 51. I²C Write Data Command

I²C WRITE DATA COMMAND	
Sequencer Command	E3h
Typical Usage	Used when a start has previously been issued with a Start sequencer command. This command writes a minimum of 1 byte, up to a maximum of 256 bytes, to an I²C slave and typically is completed with a Stop sequencer command.
I²C Features	Write Data
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the sequence (i.e., during the execution of the formed packets in the SRAM), and SPU must be active.

Formed 1-Wire Packet

Command E3h	Write Length	Write Data

Write Length

Defines the number of data bytes to write, ranging from 1 byte up to 256 bytes. Set this field to 0 to write 256 bytes.

Write Data

DATA[*n*]: Array of bytes of length *n* to write to the I²C bus. If the write length is 0, n = 256, this array must be 256 bytes.

Expected I²C Transaction

DATA[*0*]	ACK	…	DATA[*n-1*]	ACK

I^2C Read Data Command

Table 52. Read Data Command

I^2C READ DATA COMMAND	
Sequencer Command	D4h
Typical Usage	Used when a Start and I^2C Address have previously been issued, followed by a repeated start. This reads 1 to 256 bytes from an I^2C slave in one transaction and typically completes with an I^2C Stop sequencer command.
I^2C Features	Read Data
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the sequence (i.e., during the execution of the formed packets in the SRAM), and SPU must be active.

Formed 1-Wire Packet

Command D4h	Read Length	Read Array

Read Length

Defines number of data bytes to be read, ranging from 1 byte up to 256 bytes. Set read length to 0 to read 256 bytes.

Read Array

DATA[*n*]: Receive *n* bytes from the I^2C bus. Each entry in the read array, *DATA[n]*, should be set to FFh when creating the 1-Wire packet. Data received from the I^2C Read operation is stored in this array.

Expected I^2C Transaction

DATA[0]	ACK	…	DATA[*n-1*]	ACK

I2C Read Data with NACK End Command

Table 53. I2C Read Data with NACK End Command

I2C READ DATA WITH NACK END COMMAND	
Sequencer Command	D3h
Typical Usage	Used when a Start and I2C Address have previously been issued, followed by a Repeated START. This is used to read a minimum of 1 byte up to 256 bytes from an I2C slave in one transaction and should be completed with a Stop sequencer command.
I2C Features	Read Data with NACK at the end
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the sequence (i.e., during the execution of the formed packets in the SRAM), and SPU must be active.

Formed 1-Wire Packet

Command D3h	Read Length	Read Array

Read Length

Number of data bytes to read, from 1 to 256 bytes. Set the read length field to 0 to read 256 bytes.

Read Array

DATA[*n*]: A byte array of *Length to Read* size, each entry is set to FFh. Data read from the I2C device is written to this array as it is received.

Expected I2C Transaction

DATA[*0*]	ACK	DATA[*1*]	…	DATA[*n-1*]	NACK

SPI Sequencer Commands

In the beginning, the SPI sequencer will most commonly require enabling access to the SPI slave device. This is usually accomplished by issuing an SS_LOW sequencer command to make the slave select (SS) pin active low. Then an SPI Write/Read Byte or a SPI Write/Read Bit command should follow to send/receive SPI data. When the transaction has completed, an SS_HIGH sequencer command is commonly issued to restore the SS pin back to active high so as to deselect the SPI slave device.

SPI Write/Read Byte(s) Command

Table 54. SPI Write Read Byte Command

SPI WRITE/READ BYTE COMMAND	
Sequencer Command	C0h
Typical Usage	This is used first to write from 1 to 255 bytes to an SPI slave, followed by reading from 1 to 255 bytes from an SPI slave. If Write Length or Read Length is set to 0, then the Write or Read is skipped and the subsequent write data or read data field must be omitted.
SPI Features	Write, Read, or Write and Read Data using the SPI interface.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the sequence (i.e., during the execution of the formed packets in the SRAM), and SS must be active to select the SPI slave. SPU must be active to select the 1-Wire.

Formed 1-Wire Packet

Command C0h	Write Length	Read Length	Write Array[*n*]	Read Array[*m*]

Write Length

n: Number of bytes to write from 1 to 255; Setting this field to 0 indicates that no write is performed and the Write Array field must be omitted.

Read Length

m: Number of bytes to be read from 1 to 255. Setting this field to 0 indicates no read is performed and the Read Array field must be omitted.

Write Array: WDATA[*n*]

WDATA[*n*]: Array of data to write of *n* bytes as set in the Write Length field. If *n* is 0, the Write Array must not be included in the command.

Read Array: RDATA[*m*]

RDATA[*m*]: An array for storing the bytes read from the SPI bus of size *m* bytes as set in the Read Length field. The array size must be *m* bytes in size and initialized to FFh. If *m*, Read Length, is 0, the Read Array must not be included in the command.

SPI Write/Read Bit(s) Command

Table 55. SPI Write/Read Bit Command

SPI WRITE/READ BIT COMMAND	
Sequencer Command	B0h
Typical Usage	Write and/or Read from 1 to 64 bits on the SPI bus. This command enables SPI support for data and addressing in sizes other than 8 bits. For standard 8-bit communication, the SPI Write/Read Byte command is more efficient since this command is limited to 134kHz (max).
SPI Features	Write and Read or just Write or Read bits from a length of 1 to 64 bits. This command enables support for SPI data/address lengths not 8 bits in width.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the sequence (i.e., during the execution of the formed packets in the SRAM) and SS must be active to select the SPI slave. SPU must be active to select 1-Wire.

Formed 1-Wire Packet

Command B0h	Write Length	Read Length	Write Bit Array	Read Bit Array

Write Length

n: The number of data bits to write from 1 to 64. If this field is set to 0, the Write GAP is skipped, and the write data array should not be transmitted from the host.

Read Length

m: Set to the number of bits to read from 1 to 64. If this field is set to 0, the Read GAP is skipped, and no Read Bit Array should be included in the packet from the host.

Write Bit Array: WBIT[*n*]

WBIT[*n*]: Send the user-defined write data from 1 to 64 bits on byte boundaries. Only the specified number of bits are transmitted.

Read Bit Array: RBIT[*m*]

RBIT[*m*]: An array used to store the bits for the Read operation. This array must be sized on byte boundaries from 1 byte to 8 bytes in size. If the Read Length is 0, the Read Bit Array should not be used. The Read Bit Array must be initialized to FFh.

SS_HIGH Command

Table 56. SS_HIGH Command

SS_HIGH COMMAND	
Sequencer Command	01h
Typical Usage	Use this to assert an SS to the high state.
SPI Features	SS sets high logic level.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command and SPU must be active.

Formed 1-Wire Packet

Command 01h

SS_LOW Command

Table 57. SS_LOW Command

SS_LOW COMMAND	
Sequencer Command	80h
Typical Usage	Use this to de-assert the SS output pin to the low state.
SPI Features	SS sets to low logic level.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command and SPU must be active.

Formed 1-Wire Packet

Command 80h

Sequencer Utility Commands

GPIO_CTRL Write Command

Table 58. GPIO_CTRL Write Command

GPIO_CTRL WRITE COMMAND	
Sequencer Command	E2h
Typical Usage	Write the GPIO_CTRL register, configuring the GPIO pins that are not in use by the I^{2}C interface.
Typical Usage	Configure the GPIO pins for pullup enable, pullup strength, output enable, and output state.
Restriction	Only the GPIO pins not in use by the I^{2}C interface are configurable. No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command E2h	GPIO_CTRL_HI	GPIO_CTRL_LO

GPIO_CTRL_HI Byte Parameter

See Table 30 for bit descriptions.

GPIO_CTRL_LO Byte Parameter

See Table 33 for bit descriptions.

GPIO_CTRL Read Command

Table 59. GPIO_CTRL Read Command

GPIO_CTRL READ COMMAND	
Sequencer Command	2Eh
Typical Usage	Read the GPIO_CTRL register settings. **Note:** The GPIO_CTRL_HI byte and GPIO_CTRL_LO byte should be set to FFh when forming this command.
Features	Read the GPIO_CTRL register setting.
Restriction	The GPIO_CTRL register settings are only valid for GPIO pins that are not in use by the I²C interface. No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command 2Eh	GPIO_CTRL_HI Byte Set to FFh when forming this command	GPIO_CTRL_LO Low Byte Set to FFh when forming this command

GPIO_CTRL_HI Byte Parameter

See Table 30 for bit descriptions.

GPIO_CTRL_LO Byte Parameter

See Table 33 for bit descriptions.

GPIO_BUF Write Command

Table 60. GPIO_BUF Write Command

GPIO_BUF WRITE COMMAND	
Sequencer Command	D1h
Typical Usage	Write a configuration byte to the GPIO_BUF register.
GPIO Features	Isolate for low current when not used or enable the GPIOA or GPIOB input buffer when used.
Restriction	This function only effects GPIO pins not used by the I²C interface. This function has no effect if the DS28E18 is set to SPI mode. No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command D1h	GPIO_BUF 50h (BUFIZ || RFU)

GPIO_BUF Value Parameter

This writes to the GPIO_BUF parameters per Table 61 bit description.

Table 61. GPIO Buffer Register Sequencer Setting Byte - GPIO_BUF

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BUFIZ				Don't Care 'X'			

Note: See Table 32 for mapping to pin names.

BUFIZ (Bit [n]): Input Buffer Enable

0b: Input buffer is isolated (high-Z).

1b: Input buffer is enabled.

Note: The BUFIZ bits control the input buffers outside of the device function command and should be kept at default (0). The input buffers are forced on during device function commands. Therefore, make sure to turn on a pullup/pulldown or provide an external pullup per Table 31. This avoids any chance of excess crowbar current for the condition when the associated GPIO or SDA/SCL is at mid-rail or floating. See the *Power-Up of GPIO/I²C Pins* section for more details.

GPIO_BUF Read Command

Table 62. GPIO_BUF Read Command

GPIO_BUF READ COMMAND	
Sequencer Command	1Dh
Typical Usage	Read the GPIO_BUF register, which contains the input state for the GPIO pins not in use by I^2C.
GPIO Features	Read the critical nibbles of GPIO_BUF_HI/GPIO_BUF_LO register, which contains the bits to know if the input buffer is enabled or set to isolation. Additionally, when the input buffer is enabled, the input logic state of the individual GPIO pins is provided.
Restriction	This function only returns values for the GPIO pins that are not used by the I^2C interface. This function has no effect if the DS28E18 is set to SPI mode. No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command 1Dh	GPIO_BUF Value Set to FFh when forming this command

GPIO_BUF Value Parameter

This reads to the GPIO_BUF parameters per the Table 63 bit description.

Table 63. GPIO Read Buffer Register Sequencer Setting Byte – GPIO_BUF

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BUFIZ				BUF			

Note: See Table 32 for mapping to pin names.

BUFIZ (Bit [n]): Input Buffer Enable

0b: Input buffer is isolated (high-Z).

1b: Input buffer is enabled.

BUF(Bit [n]): Input Buffer

0b: Read Input buffer is 0.

1b: Read Input buffer is 1.

Delay Command

Table 64. Delay Command

DELAY COMMAND	
Sequencer Command	DDh
Typical Usage	Provides time for the I^2C slave to process (e.g., convert temp, write/read to flash, etc.)
I^2C/SPI Features	Delay function, commonly used between and I^2C/SPI command.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command DDh	Delay Parameter

Table 65. Delay Parameter

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU				DELAY			

RFU (Bit 7:4)

0b0000-0b1111: Reserved for future use.

DELAY: (Bits 3:0)

0b0000-0b1111: Delay time as specified by the following equation. The DELAY parameter is from a minimum value of 0 (0x0) to a maximum value of 15 (0xF), and the actual delay time is from 1ms to 32s respectively.

$$Delay(ms) = 2^{DELAY}$$

SENS_VDD On Command

Table 66. SENS_VDD On Command

SENS_VDD ON COMMAND	
Command Code	CCh
Typical Usage	Use this to enable the SENS_VDD output power supply for powering external I²C/SPI slaves.
I²C/SPI Features	Turns on the SENS_VDD output supply to power external I²C or SPI devices.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active. SENS_VDD automatically returns to high-Z upon completion of the command duration. Sequencer delays may need to be inserted to extend the sequencer communication time, to supply power to the I²C/SPI slave.

Formed 1-Wire Packet

Command CCh

SENS_VDD Off Command

Table 67. SENS_VDD Off Command

SENS_VDD OFF COMMAND	
Sequencer Command	BBh
Typical Usage	Use this to disable the SENS_VDD output power supply.
I²C/SPI Features	Turns off the SENS_VDD output supply, powering off any I²C/SPI devices using SENS_VDD as their supply.
Restriction	No 1-Wire commands or 1-Wire data are accepted during execution of the command, and SPU must be active.

Formed 1-Wire Packet

Command BBh

I²C

Overview

The I²C bus uses a data line (SDA) plus a clock signal (SCL) for communication. Both SDA and SCL are bidirectional lines connected to a positive supply voltage through a pullup resistor. When there is no communication, both lines are high. The output stages of devices connected to the bus must have an open drain or open collector to perform the wired-AND function. Data on the I²C bus can be transferred at rates of up to 100kbps in standard mode, up to 400kbps in fast mode, and up to 1Mbps in Fm+. A device that sends data on the bus is defined as a transmitter, and a device receiving data is defined as a receiver. The device that controls the communication is called a master. The devices that are controlled by the master are slaves. To be individually accessed, each device must have a slave address that does not conflict with other devices on the bus. Data transfers can be initiated only when the bus is not busy. The master generates the serial clock (SCL), controls the bus access, generates the START and STOP conditions, and determines the number of data bytes transferred between START and STOP, as seen in Figure 9. Data is transferred in bytes, with the most significant bit being transmitted first. After each byte follows an acknowledge bit to allow synchronization between master and slave.

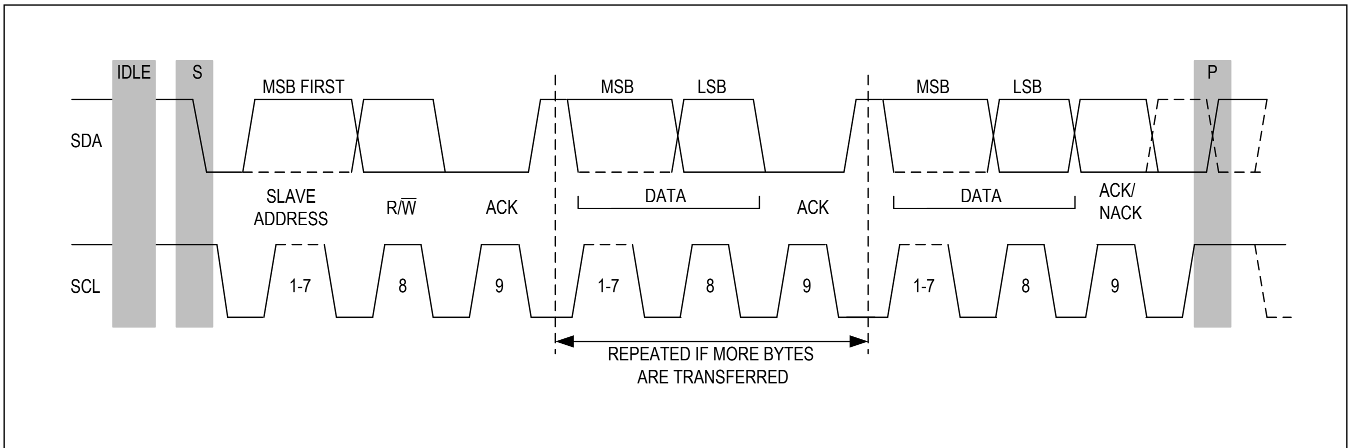


Figure 9. I²C Protocol Overview

I²C Definitions

The following terminology is commonly used to describe I²C data transfers. The timing references are defined in Figure 10.

Bus Idle or Not Busy

Both SDA and SCL are inactive and in their logic-high states.

START Condition

To initiate communication with a slave, the master must generate a START condition. A START condition is defined as a change in state of SDA from high to low while SCL remains high.

STOP Condition

To end communication with a slave, the master must generate a STOP condition. A STOP condition is defined as a change in state of SDA from low to high while SCL remains high.

Repeated START Condition

Repeated STARTs are commonly used for read accesses after having specified a memory address to read from in a preceding write access. The master can use a repeated START condition at the end of a data transfer to immediately initiate a new data transfer following the current one. A repeated START condition is generated the same way as a normal

START condition, but without leaving the bus idle after a STOP condition.

Data Valid

With the exception of the START and STOP condition, transitions of SDA can occur only during the low state of SCL. The data on SDA must remain valid and unchanged during the entire high pulse of SCL plus the required setup and hold time ($t_{HD:DAT}$ after the falling edge of SCL and $t_{SU:DAT}$ before the rising edge of SCL; see Figure 10). There is one clock pulse per bit of data. Data is shifted into the receiving device during the rising edge of the SCL pulse.

When finished with writing, the master must release the SDA line for a sufficient amount of setup time (minimum $t_{SU:DAT}$, + t_R in Figure 10) before the next rising edge of SCL to start reading. The slave shifts out each data bit on SDA at the falling edge of the previous SCL pulse, and the data bit is valid at the rising edge of the current SCL pulse. The master generates all SCL clock pulses, including those needed to read from a slave.

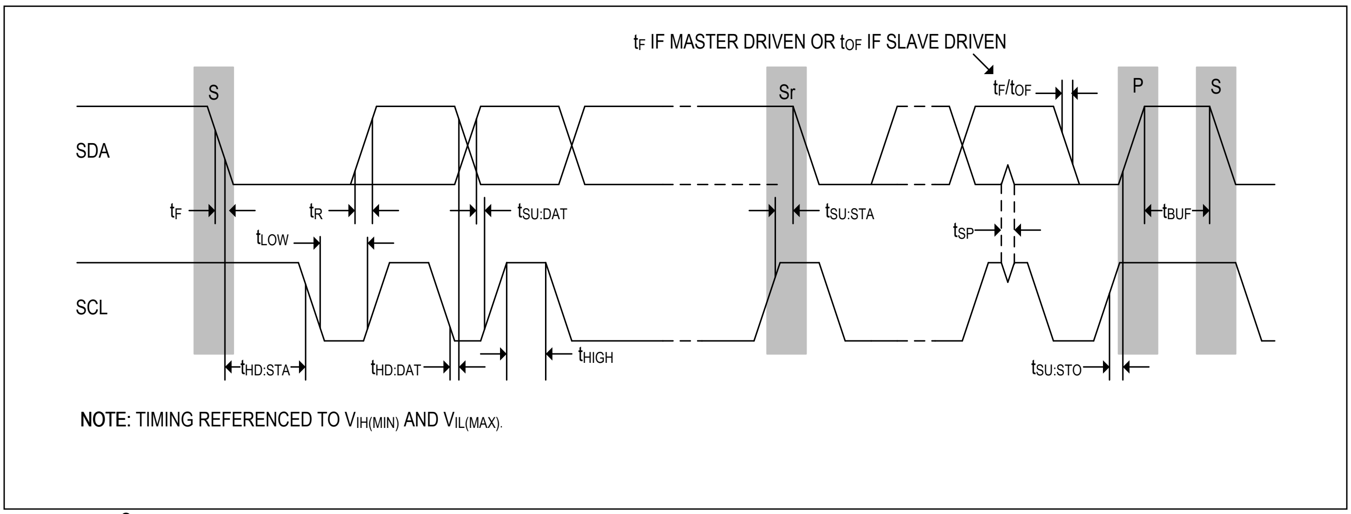


Figure 10. I²C Timing Diagram

SPI

Overview

Serial peripheral interface (SPI) is a 4-wire, synchronous serial communication bus used for short-distance communication. SPI devices communicate in full duplex mode using a master-slave architecture with a single master. The SPI master device controls the communication frame for reads and writes using the slave select output. The DS28E18 SPI master supports a single slave select line, allowing communication between the DS28E18 and a single slave SPI device. For single slave SPI networks, the Slave Select pin (SS#) is an output only and defaults to active low. See Figure 11 for additional details.

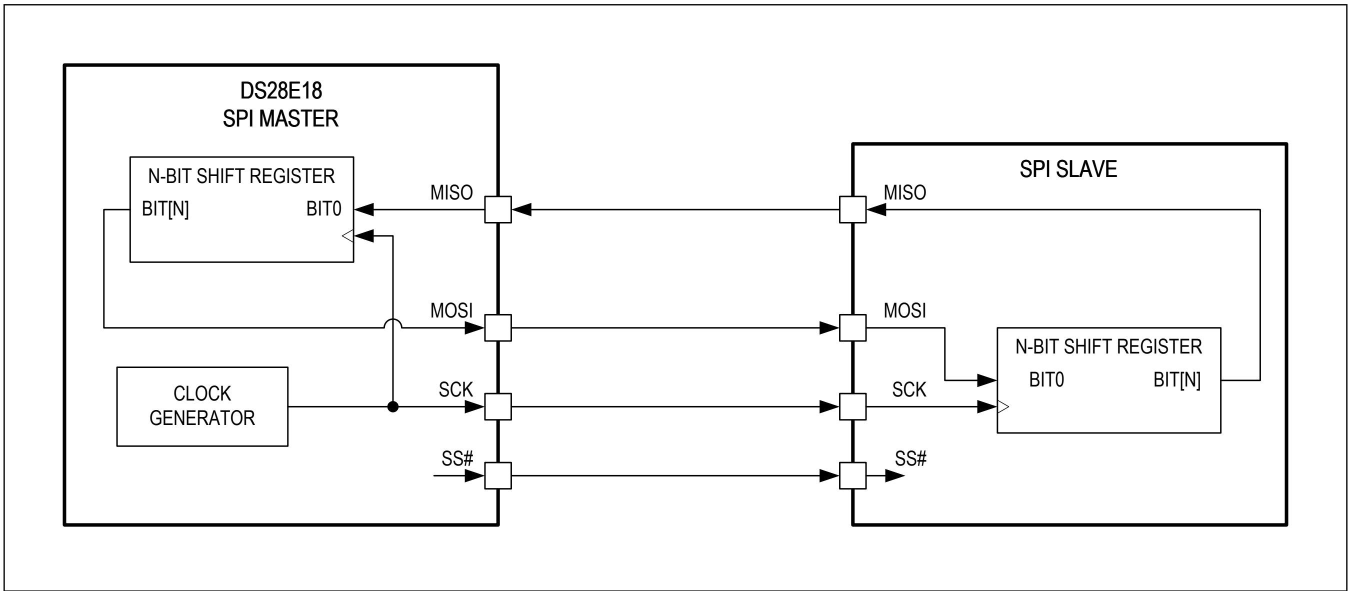


Figure 11. SPI Single Slave Network

SPI Timing

The timing references are defined in Figure 12.

SPI Timing Diagram

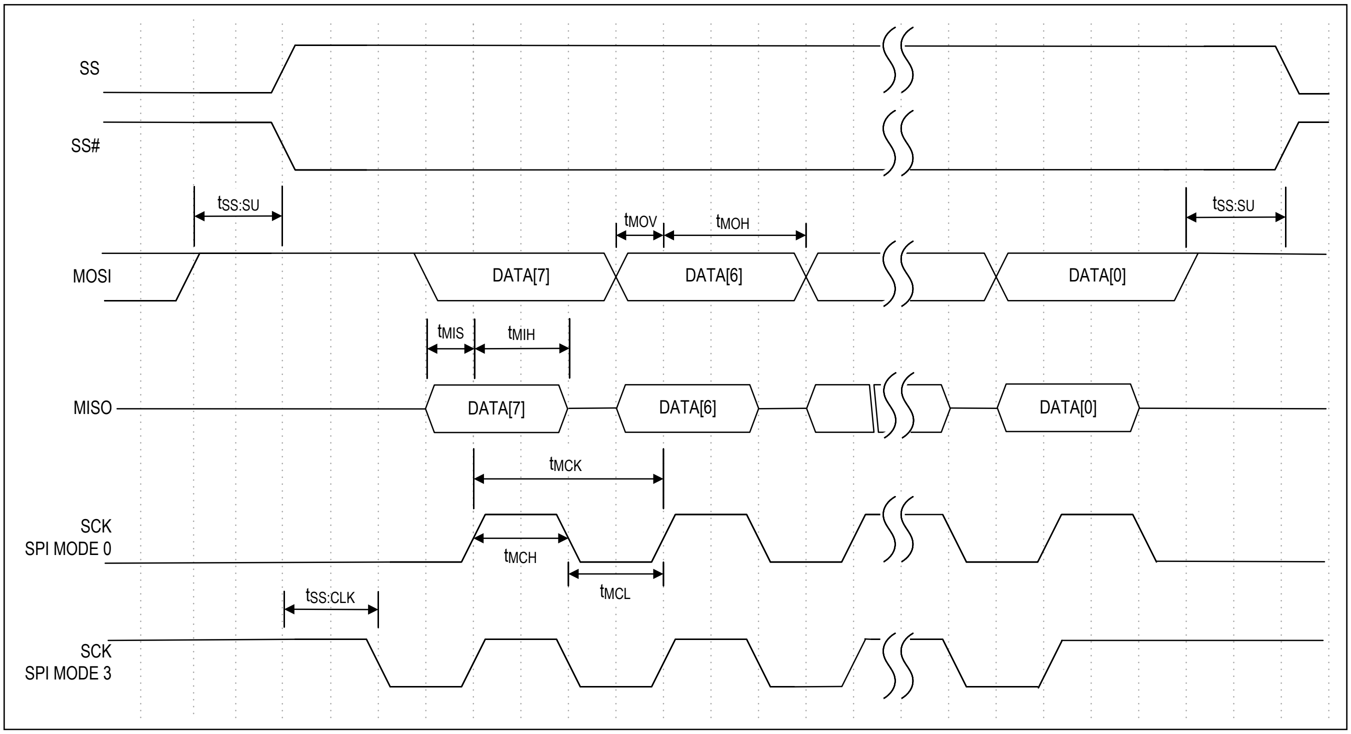


Figure 12. SPI Timing Diagram

Power-Up of GPIO/I²C Pins

The device function commands enable the GPIOA, GPIOB, SCL, and SDA input buffers regardless of the BUFIZ register setting. The BUFIZ register should be kept at default (0) to disable the input buffers outside of the device function command duration. Care should be taken to turn on a pullup/pulldown or provide an external pullup. This avoids any chance of excess internal crowbar current occurring by preventing GPIOA/GPIOB or SDA/SCL being at a mid-rail or floating condition.

An example configuration to avoid the issue is shown in Table 68. After POR, this example sequence sets GPIOA/GPIOB with a 25kΩ internal pullup resistor and SCL/SDA with a 2.7kΩ internal pullup resistor, which will prevent a mid-rail or floating condition on the pins.

Table 68. Example Write GPIO Configuration Sequence after POR

Reset
Presence Pulse
<ROM Select>
Tx: Command Start (66h)
Tx: Length Byte (5)
Tx: Command 83h (write GPIO configuration)
Tx: Parameter (CFG_REG_TARGET = 0Bh)
Tx: Parameter (CFG_REG_MOD = 03h)
Tx: Parameter (GPIO_CTRL_HI = A5h)
Tx: Parameter (GPIO_CTRL_LO = 0Fh)
Rx: CRC-16 (inverted of command start, length, command, and all parameters)
Tx: Release Byte (AAh)
<Delay t_{OP}>
Rx: Dummy Byte (not used in CRC calculation) = FFh
Rx: Length Byte (1)
Rx: Result Byte = AAh expected
Rx: CRC-16 (inverted of length and result byte)
Reset

Timeout

In I²C mode or SPI mode, the internal master can time out if something external to the DS28E18 holds the bus in a way that prevents the intended transmitted communication from being generated. This timeout will occur nominally within 6µs after the completion of the transmitted event. For SPI mode, timeout is only applicable when using the SPI Write/Read Byte sequencer command.

A few examples that can cause a timeout:

- SCL is held high or held low during a Write/Read Data sequencer command being executed.
- SDA is held high during a Start.
- SDA is held high during a Stop.
- SS# pin is held high during a SS# high to low transition.

Contact the factory if you need to disable this feature.

Typical Application Circuits

DS28E18 Configured as an I2C Master

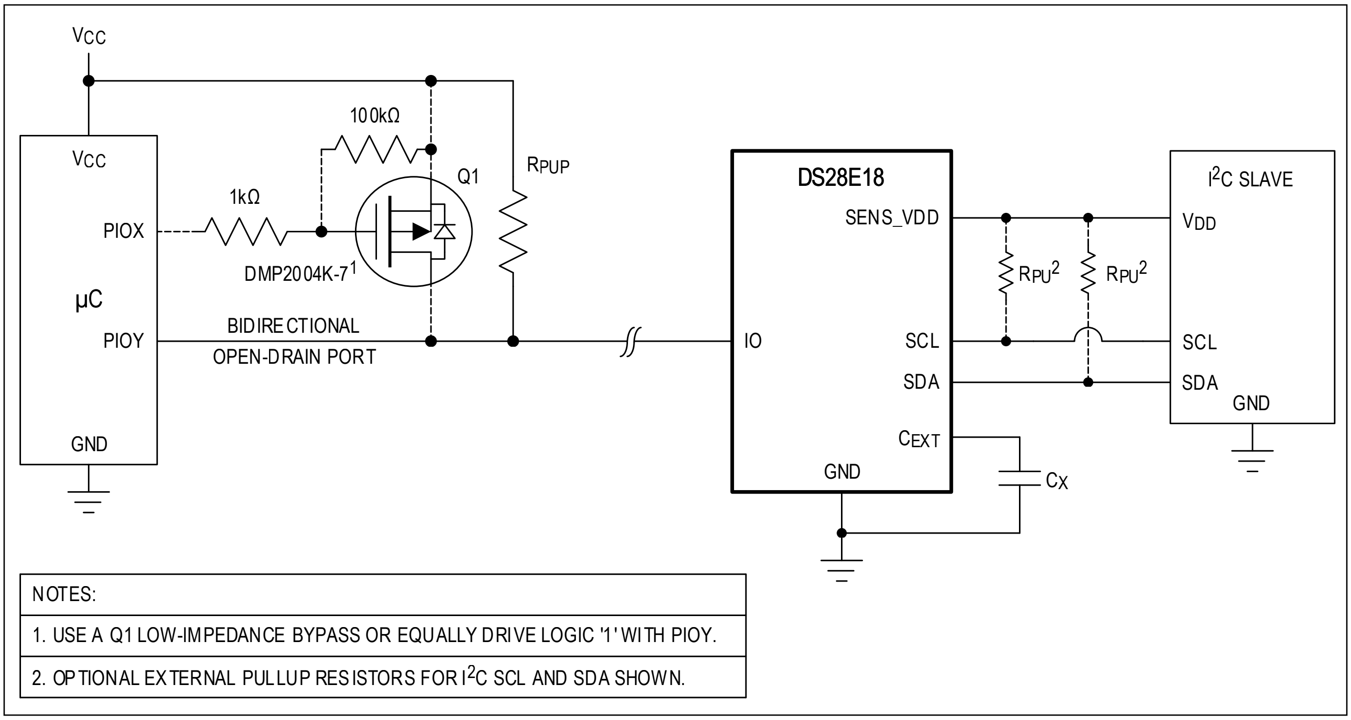


Figure 13. DS28E18 Configured as an I2C Master

Typical Application Circuits (continued)

DS28E18 Configured as an SPI Master

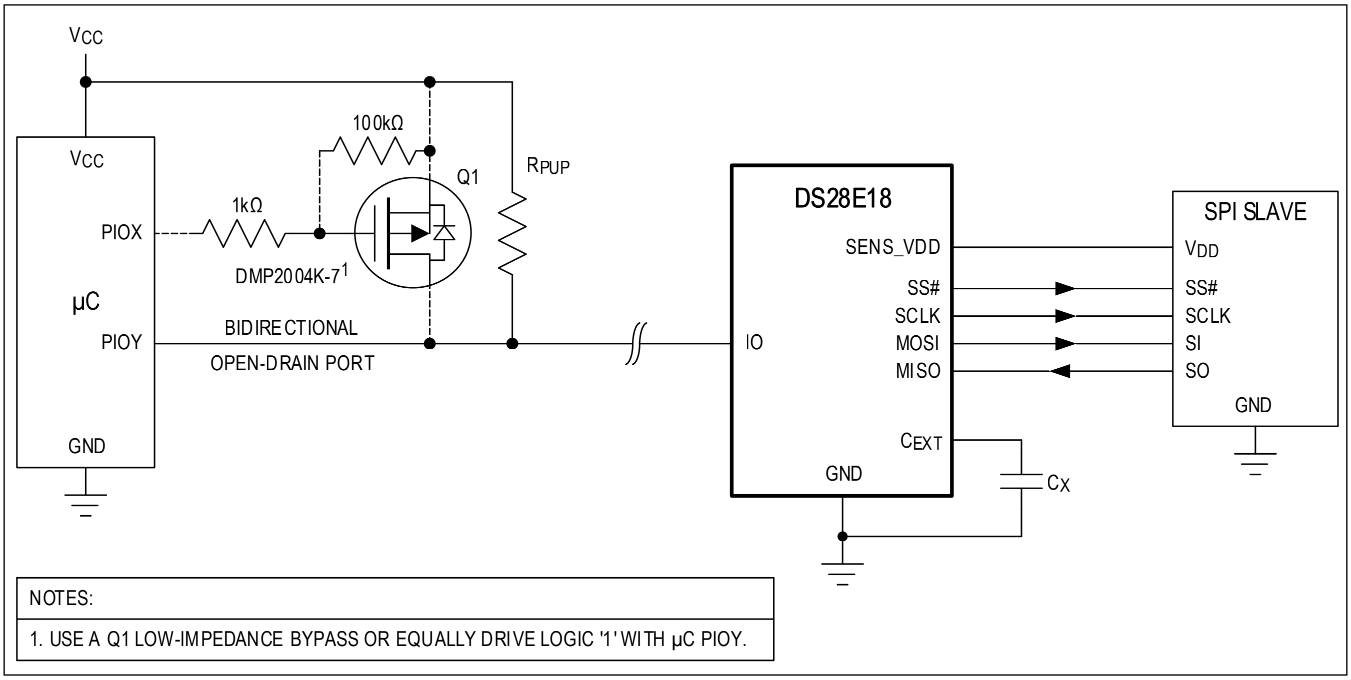


Figure 14. DS28E18 Configured as an SPI Master

Ordering Information

PART NUMBER	TEMP RANGE	PIN-PACKAGE
DS28E18Q+T	-40°C to +85°C	8 TDFN-EP*

+Denotes a lead(Pb)-free/RoHS-compliant package.

T = Tape and reel.

**EP = Exposed pad.*

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	6/20	Release for market intro	—
1	1/24	Updated Electrical Characteristics and Table 27	3, 35